Weak Prognosability of Discrete Event Systems*

Shaowen Miao[1], Petr Jančar[2], Jan Komenda[3], Tomáš Masopust[2], Aiwen Lai[4], and Yiding Ji[1]

***Abstract*— Prognosability is a property that assesses whether faults can be predicted in advance based on a limited number of observation steps. However, it is inherently restrictive, as it requires *all* fault occurrences to be predictable. In large-scale and complex systems, certain fault scenarios are often intrinsically unpredictable due to limited sensor coverage, whereas others are well monitored. As a result, classical prognosability fails to capture the partial predictability exhibited by such systems. In this paper, we investigate a weaker version of prognosability, termed *weak prognosability*, which fills this gap by characterizing systems that can predict faults in specific scenarios (i.e., along certain strings), while distinguishing them from behaviors that are completely opaque to fault prediction. Specifically, we study the computational complexity and decidability of checking weak prognosability for automata and labeled Petri nets. We show that deciding weak prognosability is PSPACE-complete for automata, EXPSPACE-complete for modular automata, and undecidable for labeled Petri nets.**

***Index Terms*— discrete event systems, weak prognosability, automata, Petri nets, decidability, computational complexity.**

I. INTRODUCTION

The verification of diagnosability for discrete-event systems (DESs) has been a commonplace topic, attracting considerable attention over the recent decades [1], [2]. However, a potential drawback of diagnosability is that it requires the detection of a fault only after it has occurred. While early detection of faults in practical systems can mitigate losses, there remains the possibility of incurring unforeseen damage.

To address this, Genc and Lafortune [3] introduced the concept of prognosability (or predictability). Unlike diagnosability, prognosability requires that any fault must be predicted with certainty before its actual occurrence, based on certain observations. Interestingly, prognosability for nondeterministic finite automata (NFAs) can be verified using verifier [4], meaning its verification complexity is no worse than that of diagnosability. However, Haar et al. [5] demonstrated that synthesizing control to enforce predictability is EXPTIME-complte, which is significantly more complex the PSPACE-complete problem of control for diagnosability [6].

Diagnosability and prognosability have been extended to various formalisms. For large-scale systems composed of interacting modules, research has expanded to modular DESs [7], [8] and epistemic settings [9]. Also works like [10], [11], [12], [13] explored prognosability in probabilistic settings.

Verification of diagnosability [14], [15] and prognosability [16], [17], [18], [19] has also been extensively explored for labeled Petri nets (LPNs). [16] proposed a sufficient condition to verify prognosability in both bounded and unbounded partially observed Petri nets (PNs). [17] has shown that deciding prognosability for unbounded LPNs is EXPSPACE-complete. [18] constructed a predictor graph to verify the prognosability of both bounded and unbounded LPNs.

Despite these theoretical advances, the standard definition of prognosability imposes a stringent "all-or-nothing" requirement: a system is prognosable only if *every* possible fault trajectory can be predicted. In practical scenarios, this requirement is often overly conservative. Complex systems may possess limited sensor coverage due to cost or physical constraints, rendering certain fault scenarios intrinsically unpredictable while others remain observable. According to the standard definition, such a system would be deemed *not prognosable*, ignoring its partial predictive capabilities. As pointed out by [20] in the context of diagnosis, obtaining more system observations typically necessitates prohibitive sensor costs. This realization has led to the emergence of weak diagnosability [20]. In a similar spirit, this paper introduces the concept of weak prognosability and investigates its verification complexity using (modular) NFAs and LPNs.

We argue that in systems where global predictability is unattainable, it is still valuable to verify whether specific, critical fault scenarios are predictable. Our main theoretical contributions are outlined as follows: we introduce the concept of weak prognosability for NFAs and show that deciding weak prognosability is PSPACE-complete for monolithic NFAs, EXPSPACE-complete for modular NFAs, and undecidable for unbounded LPNs.

The remainder is structured as follows. Section II provides a brief overview of the preliminaries utilized in this study. Section III explores the computational complexity of weak prognosability for (modular) NFAs. Section IV shows the undecidability of weak prognosability for LPNs. Section V concludes the paper and outlines future research directions.

*Financial support: National Natural Science Foundation of China grants 62303389, 62373289; Guangdong Basic and Applied Basic Research Funding grant 2024A1515012586; Guangdong Scientific Research Platform and Project Scheme grant 2024KTSCX039; Youth Science and Technology Talent Support Program of Guangdong Provincial Association for Science and Technology grant SKXRC2025463; Guangdong Provincial Key Lab of Integrated Communication, Sensing and Computation for Ubiquitous Internet of Things grant 2023B1212010007; Fujian Provincial Natural Science Foundation of China Grant 2025J01054; Natural Science Foundation of Xiamen, China grant 3502Z202573035; Fundamental Research Funds for the Central Universities grant 20720250160; and RVO 67985840.

[1]S. Miao and Y. Ji are with Robotics and Autonomous Systems Thrust, Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China. smiao585@connect.hkust-gz.edu.cn, jiyiding@hkust-gz.edu.cn (Corresponding author: Yiding Ji.)

[2]P. Jančar and T. Masopust are with the Faculty of Science, Palacky University Olomouc, Czechia petr.jancar@upol.cz, tomas.masopust@upol.cz

[3]J. Komenda is with the Institute of Mathematics of the Czech Academy of Sciences, Prague, Czechia komenda@ipm.cz

[4]A. Lai is with the Department of Automation, Xiamen University, Xiamen 361102, China aiwenlai@xmu.edu.cn

II. Preliminaries and Definitions

Let $\mathbb{N}$ denote the set of non-negative integers. For a set A, the notation $|A|$ stands for the cardinality of A. An alphabet Σ is a finite nonempty set of labels. A word over Σ is a finite sequence of labels from Σ. Let Σ^* denote the set of all words over Σ, including the empty word ε. For a word $w \in \Sigma^*$, we use the notation $|w|$ to denote the length of w, and $\overline{w}$ to denote the set of all prefixes of w. For two words $w_1, w_2 \in \Sigma^*$, we denote by $w_1 \cdot w_2$ (or simply $w_1 w_2$) the *concatenation* of w_1 and w_2. A language over Σ is a subset $L \subseteq \Sigma^*$. The *post-language* (or left quotient) of L after a word $w_1 \in L$ is the set $L/w_1 = \{w_2 \in \Sigma^* \mid w_1 w_2 \in L\}$.

A. Complexity Theory

Here, we review the necessary concepts of complexity and computability theory [21]. A *(decision) problem* is a yes-no question referring to problem instances. A problem is *decidable* if there is an algorithm that decides it. In complexity theory, decidable problems are classified into different classes based on the time or space required by an algorithm to solve them. We denote by PSPACE and EXPSPACE the classes of problems solvable by deterministic polynomial-space and deterministic exponential-space algorithms, respectively. For $X \in \{\text{PSPACE, EXPSPACE}\}$, a problem is X-complete if (i) it belongs to X and (ii) every problem from X can be reduced to it in deterministic polynomial time. Condition (i) is known as *membership* and (ii) as *hardness*. The inclusion PSPACE $\subsetneq$ EXPSPACE is widely acknowledged and it is believed that no polynomial-time algorithms exist for PSPACE-complete problems.

B. Nondeterministic Finite Automata (NFAs)

We recall the basic notions of automata theory [22]. An *NFA* is a tuple $G = (Q, \Sigma, \delta, I, F)$, where Q is a finite nonempty set of states, $I \subseteq Q$ is a nonempty set of initial states, $F \subseteq Q$ is a set of accepting states, and $\delta\colon Q \times \Sigma \to 2^Q$ is a transition function, extendable to $\delta\colon 2^Q \times \Sigma^* \to 2^Q$ by induction. The language *generated* by G is defined by $L(G) = \{w \in \Sigma^* \mid \delta(I, w) \neq \emptyset\}$, and the language *accepted* by G is defined by $L_m(G) = \{w \in \Sigma^* \mid \delta(I, w) \cap F \neq \emptyset\}$. If F is irrelevant, we omit it and write $G = (Q, \Sigma, \delta, I)$. An NFA is *total* if its transition function δ is total, i.e., $\delta(q, a) \neq \emptyset$ for every $q \in Q$ and $a \in \Sigma$.

In a *partially observed* NFA G, its alphabet Σ is partitioned into *observable* events Σ_o and *unobservable* events Σ_{uo}. The *projection* $P\colon \Sigma^* \to \Sigma_o^*$ is a morphism for concatenation defined by $P(a) = a$ for $a \in \Sigma_o$, and $P(a) = \varepsilon$ for $a \in \Sigma_{uo}$; hence $P(a_1 \cdots a_n) = P(a_1) \cdots P(a_n)$ for every word $a_1 \cdots a_n \in \Sigma^*$. The *inverse projection* of P is defined by $P^{-1}(w) = \{v \in \Sigma^* \mid P(v) = w\}$. The definitions can naturally be extended to languages. By $P(G)$, we denote the NFA obtained from G by replacing every transition (p, a, q) of G with $(p, P(a), q)$; intuitively, $P(G)$ has the same structure as G with unobservable transitions labeled with ε. A modular NFA G is a set $\{G_1, \ldots, G_n\}$ of NFAs, where $n \geq 2$, the behavior of which coincides with the parallel composition $\|_{i=1}^n G_i$. An NFA is *live* (or *deadlock-free*) if there is an outgoing transition from every reachable state, and it is *convergent* if there is no infinite sequence of unobservable events from any reachable state.

C. Labeled Petri Nets (LPNs)

We recall some basics of LPNs [23]. A *Petri Net (PN)* is a tuple $N = (P, T, \mathrm{Pre}, \mathrm{Post})$, where P is a set of m places, T is a set of n transitions with $P \cup T \neq \emptyset$ and $P \cap T = \emptyset$, and $\mathrm{Pre}\colon P \times T \to \mathbb{N}$ and $\mathrm{Post}\colon P \times T \to \mathbb{N}$ are the pre- and post-incidence functions specifying the arcs directed from places to transitions and from transitions to places, respectively. A *marking* is a map $M\colon P \to \mathbb{N}$ assigning to each place a number of tokens. A *PN system* (N, M_0) consists of a PN N and an initial marking M_0. A transition t is *enabled* in M if $M(p) \geq \mathrm{Pre}(p, t)$ for every place $p \in P$. The *firing* of an enabled transition t in M leads to the marking M', where $M'(p) = M(p) - \mathrm{Pre}(p, t) + \mathrm{Post}(p, t)$ for every $p \in P$.

We use the notation $M \xrightarrow{\sigma}$ to denote that the sequence of transitions $\sigma \in T^*$ is enabled in M, and $M \xrightarrow{\sigma} M'$ to denote that firing the sequence σ results in the marking M'. Let $L(N, M_0) = \{\sigma \in T^* \mid M_0 \xrightarrow{\sigma}\}$ denote the set of all transition sequences that can fire from the initial marking M_0. A marking M is reachable in (N, M_0) if there exists a transition sequence $\sigma \in T^*$ such that $M_0 \xrightarrow{\sigma} M$.

An *LPN* is a tuple $G = (N, M_0, \Sigma, \ell)$, where (N, M_0) is a PN system, Σ is an alphabet, and $\ell\colon T \to \Sigma \cup \{\varepsilon\}$ is a function assigning labels to transitions. The labeling function can be extended to $\ell\colon T^* \to \Sigma^*$ by $\ell(\sigma t) = \ell(\sigma)\ell(t)$ for $\sigma \in T^*$ and $t \in T$. A transition t is observable if $\ell(t) \in \Sigma$, and unobservable if $\ell(t) = \varepsilon$. The *inverse image* of ℓ, denoted as ℓ^{-1}, is defined in a usual way. The language generated by G is the set $L(G) = \{\ell(\sigma) \in \Sigma^* \mid \sigma \in L(N, M_0)\}$.

An LPN is *live* if there is an enabled transition from every reachable marking. It is *convergent* if no infinite sequence of unobservable transitions exists from any reachable marking.

III. Deciding Weak Prognosability for NFAs

This section studies the decidability of weak prognosability for NFAs. The main contributions include: deciding weak prognosability is PSPACE-complete and deciding weak prognosability in the modular setting is EXPSPACE-complete.

A. Weak Prognosability

As is common in the literature, we consider a single type of fault, while the extension to multiple fault types can be handled on a type-by-type basis. Let $\Sigma_f \subseteq \Sigma$ denote the set of fault events (for a given type). We call any word in $\Sigma^* \Sigma_f \Sigma^*$ a *faulty* word, and all other words *non-faulty*. We first recall the definition of standard prognosability [3] and then propose the definition of weak prognosability of NFAs.

Definition 1 (Prognosability of NFAs)**.** *A live and convergent NFA G over Σ is* prognosable *with respect to projection $P\colon \Sigma^* \to \Sigma_o^*$ and fault events $\Sigma_f \subseteq \Sigma$ if*

$$(\exists K \in \mathbb{N})\,(\forall w_1 \in \Sigma^* \Sigma_f \cap L(G))\,(\exists w_2 \in \overline{w_1})[\Sigma_f \notin w_2 \wedge \mathbf{P}]$$

where $\mathbf{P} \equiv (\forall w \in L(G))(\forall w' \in L(G)/w)$

$$[(P(w) = P(w_2)) \wedge (\Sigma_f \notin w) \wedge (|w'| \geq K) \Rightarrow (\Sigma_f \in w')]\,.$$

Definition 2 (Weak Prognosability of NFAs)**.** *A live and convergent NFA G over Σ is* weakly prognosable *with respect to $P\colon \Sigma^* \to \Sigma_o^*$ and $\Sigma_f \subseteq \Sigma$ if*

$$(\exists K \in \mathbb{N})\,(\exists w_1 \in \Sigma^*\Sigma_f \cap L(G))\,(\exists w_2 \in \overline{w_1})[\Sigma_f \notin w_2 \wedge \mathbf{P}].$$

Weak prognosability requires that the system is able to predict some fault occurrences several steps in advance. A related concept was discussed in [10, Definition 13], which requires that every word from $\Sigma^*\Sigma_f \cap L(G)$ has a non-zero probability of predicting future faults. However, the notion presented in this paper involves at least one absolutely successful prediction, and may involve some faults that can never be predicted. In contrast, standard prognosability requires that the system predicts every fault occurrence.

We now provide an example illustrating the distinction between weak and standard prognosability.

Example 1. *Consider the NFA G depicted in Fig. 1. For the fault event e_f, the system is weakly prognosable; indeed, after observing a, we can predict that e_f will occur in one step. However, we cannot predict whether e_f will occur after observing b, and hence G is not prognosable.*

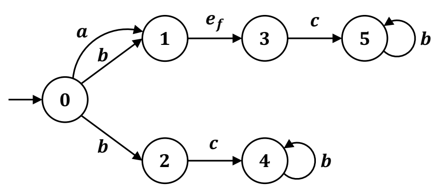

Fig. 1. An NFA $G = (Q, \Sigma, \delta, I)$, where $\Sigma_o = \{a, b, c\}$ and $\Sigma_f = \{e_f\}$.

In real-world systems, it is often impractical to predict every fault occurrence several steps in advance due to limited sensor coverage. Weak prognosability facilitates predication for selected high-priority fault scenarios, allowing for a more refined safety analysis where early warnings can be guaranteed for critical or frequent failure modes, even when full prognosability is not ensured for rare corner cases. Additionally, weak prognosability can be combined with (weak) diagnosability to ensure that all faults are detected after they occur, and some faults are predicted in advance. This combination provides a more comprehensive fault management strategy for practical systems. For completeness, we recall the definition of weak diagnosability [20].

Definition 3 (Weak Diagnosability for NFAs)**.** *A live and convergent NFA G over Σ is* weakly diagnosable *with respect to projection $P\colon \Sigma^* \to \Sigma_o^*$ and fault events $\Sigma_f \subseteq \Sigma_{uo}$ if*

$$(\forall w_1 \in L(G) \cap \Sigma^*\Sigma_f\Sigma^*)(\exists w_2 \in L(G)/w_1)$$
$$(\forall w \in L(G))\,[P(w) = P(w_1w_2) \Rightarrow (\Sigma_f \in w)]\,.$$

It is worth noting that weak prognosability and weak diagnosability are incomparable: weak prognosability does not imply weak diagnosability, and vice versa; see the following example. This observation is markedly different from the relationship between prognosability and diagnosability, where prognosability implies diagnosability [3].

Example 2. *Consider again the system G of Fig. 1. We can observe that G is not weakly diagnosable; indeed, for the word be_f with the observation b, we cannot detect the occurrence of e_f no matter what the continuation of b is. On the other hand, if we remove the transition $0 \xrightarrow{a} 1$ and replace $2 \xrightarrow{c} 4$ with $2 \xrightarrow{a} 4$, the resulting automaton becomes weakly diagnosable but not weakly prognosable.*

We now adapt the diagnoser technique of [3] for verifying prognosability to verify weak prognosability. Given an NFA G, its diagnoser is denoted by

$$\begin{aligned}\mathrm{Diag}(G) &= \mathrm{Obs}(G \| A_\ell)\\ &= (X, \Sigma_o, \xi, x_0) = \mathrm{Ac}(2^{Q\times\{N,Y\}}, \Sigma_o, \xi, x_0),\end{aligned}$$

where $A_\ell = (\{N, Y\}, \Sigma_f, \delta_\ell, \{N\})$ is the labeling automaton with the transitions $\delta_\ell(N, e_f) = \delta_\ell(Y, e_f) = \{Y\}$ for all $e_f \in \Sigma_f$, and $\mathrm{Ac}(\cdot)$ denotes the part of the automaton that is accessible from the initial state. The transition function $\xi\colon X \times \Sigma_o \to X$ is defined by $\xi(x, a) = \{q \in Q \times \{N, Y\} \mid$ there are $q' \in x$ and $v \in \Sigma_{uo}^*$ such that $q \in \delta_{G\|A_\ell}(q', va)\}$, where $\delta_{G\|A_\ell}$ is the transition function of $G\|A_\ell$. The transition function ξ can be extended to $X \times \Sigma_o^*$ as usual. A state $x = \{(q_1, \ell_1), \ldots, (q_m, \ell_m)\} \in X$, for $m \geq 1$, of $\mathrm{Diag}(G)$ is *normal* if $\ell_1 = \cdots = \ell_m = N$; it is *faulty* if $\ell_1 = \cdots = \ell_m = Y$; and it is *uncertain* if there are $j, k \in \{1, \ldots, m\}$ such that $\ell_j = N$ and $\ell_k = Y$. Let $X_N, X_F, X_U \subseteq X$ denote the sets of normal, faulty, and uncertain states, respectively. We denote by $X_{N\to} = \{x \in X_N \mid$ there is $a \in \Sigma_o$ such that $\xi(x, a) \in X_F \cup X_U\}$ the set of normal states with an outgoing transition to a faulty or uncertain state. We further use $\mathrm{Ac}(\mathrm{Diag}(G), x)$ to denote the accessible part of $\mathrm{Diag}(G)$ from the state $x \in X$.

Genc and Lafortune [3] have shown that an NFA G is prognosable if and only if for every $x \in X_{N\to}$, all cycles in $\mathrm{Ac}(\mathrm{Diag}(G), x)$ consist solely of faulty states.

Note that for every state $x \in X_{N\to}$, there exists $(q_i, N) \in x$ such that the fault event e_f is feasible in q_i, i.e., $\delta(q_i, e_f) \subseteq Q$. Therefore, combined with Definitions 1 and 2, we derive the following lemma, the proof of which can be readily adapted from Theorem 8 in [3], and is therefore omitted.

Lemma 1. *A live and convergent NFA G is weakly prognosable if and only if there exists $x \in X_{N\to}$ such that all cycles in $\mathrm{Ac}(\mathrm{Diag}(G), x)$ consist only of faulty states.*

The following example illustrates the lemma.

Example 3. *Reconsider the NFA G depicted in Fig. 1. We construct its diagnoser $\mathrm{Diag}(G)$ as shown in Fig. 2. By definition, we have $X_{N\to} = \{\{(1, N)\}, \{(1, N), (2, N)\}\}$. Since for $\{(1, N)\}$, there is a unique faulty cycle $\{(5, Y)\} \xrightarrow{b} \{(5, Y)\}$, Lemma 1 gives that G is weakly prognosable.*

Then we leverage Lemma 1 to determine the complexity of deciding weak prognosis of NFAs

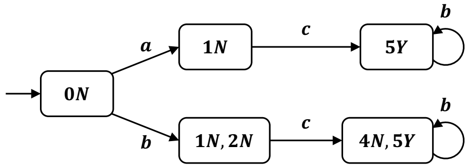

Fig. 2. The diagnoser $\mathrm{Diag}(G)$.

Theorem 1. *Deciding weak prognosability of a live and convergent NFA is PSPACE-complete.*

Proof. Let G be an NFA with state set Q, and let $\mathrm{Diag}(G)$ denote its diagnoser. Since each diagnoser state is a subset of $Q \times \{N, Y\}$, the size of each state of the diagnoser is polynomial in $|Q|$. Hence, in the transition structure of $\mathrm{Diag}(G)$, the classical nondeterministic search for checking the property of Lemma 1 can be solved in PSPACE. Notice that $\mathrm{Diag}(G)$ hides unobservable cycles, which justifies the requirement on the convergence of the NFA.

To show PSPACE-hardness, we reduce the problem of language non-universality with all states accepting [24] to the problem of weak prognosability. Given an NFA G over Σ, the language non-universality problem asks whether $L(G) \neq \Sigma^*$. Starting with an NFA $G = (Q, \Sigma, \delta, \{q_0\})$, we construct a new NFA $G' = (Q \cup \{q', q_1, q_2, q_f, q_t\}, \Sigma \cup \{\diamondsuit, e_f, u_1, u_2\}, \delta', \{q'\})$, where $q', q_1, q_2, q_f, q_t \notin Q$ are new states, events $e_f, \diamondsuit, u_1, u_2 \notin \Sigma$, u_1 and u_2 are unobservable, and e_f is a fault event. The transition function δ' is initialized as δ and extended by transitions (q', u_1, q_0), $(q, \diamondsuit, q_t)$ for every $q \in Q$, by $(q_t, \diamondsuit, q_t)$, (q', u_2, q_1), (q_1, a, q_1) for every $a \in \Sigma$, and by $(q_1, \diamondsuit, q_2)$, (q_2, e_f, q_f), $(q_f, \diamondsuit, q_f)$, see Fig. 3.

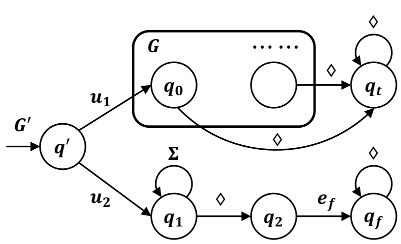

Fig. 3. Sketch of the reduction in the proof of Theorem 1.

We now show that the language $L(G)$ is not universal if and only if G' is weakly prognosable. If $L(G) = \Sigma^*$, then we consider an arbitrary natural number K and an arbitrary faulty word $u_2 w \diamondsuit e_f \in \Psi_{G'}(e_f)$. For every non-faulty prefix $u_2 v$ of $u_2 w \diamondsuit e_f$, there is a non-faulty word $u_1 v \in L(G')$ with the same projection as $u_2 v$, and its non-faulty extension $v' \diamondsuit^K$ of length at least K, where $vv' = w$. Since this argument also applies to the empty prefix, the system G' is not weakly prognosable.

On the other hand, if G is not universal, let $w \notin L(G)$. We take $K = 1$, the faulty word $u_2 w \diamondsuit e_f$ of G', and its non-faulty prefix $u_2 w \diamondsuit$. Since there is no word in $P^{-1}(P(u_2 w \diamondsuit))$ in G' starting with u_1, condition $\mathbf{P}$ of Definition 2 is satisfied. Hence G' is weakly prognosable, which completes the proof. □

B. Weak Modular Prognosability

Given a modular NFA $\{G_1, \ldots, G_n\}$ with G_i over Σ_i, a set of unobservable events $\Sigma_{uo} \subseteq \bigcup_{i=1}^n \Sigma_i$, and a fault event $e_f \in \Sigma_{uo}$. The weak modular prognosability asks whether the system $\|_{i=1}^n G_i$ is weakly prognosable with respect to $P \colon \Sigma^* \to \Sigma_o^*$ and $\{e_f\}$.

Masopust and Yin [8] have shown that deciding A-diagnosability is EXPSPACE-complete for modular NFAs by reducing the EXPSPACE-complete problem determining whether a regular expression with squaring ($s^2 = s \cdot s$) is universal [25]. It is sufficient to consider an expression E over $\Delta = \{\#\} \cup T \cup Q \times T$, where T and Q are disjoint sets, of the form

$$((\Delta \setminus \#) \cup \#) \cdot ((\Delta \setminus (q_0, x_1)) \cup (q_0, x_1) \cdot ((\Delta \setminus x_2) \cup x_2 \cdot ((\Delta \setminus x_3) \cup \cdots \cup (\Delta \setminus x_n)) \cdots) \cdot \Delta^* \quad (1)$$

$$\cup\, \Delta^{n+1} \cdot b^* \cdot (\Delta \setminus \{b, \#\}) \cdot \Delta^* \quad (2)$$

$$\cup\, \# \cdot (\Delta \cup \varepsilon)^{2^n - 1} \cdot \# \cdot \Delta^* \quad (3)$$

$$\cup\, \# \cdot \Delta^{2^n} \cdot (\Delta \setminus \#) \cdot \Delta^* \quad (4)$$

$$\cup\, (\Delta \setminus (\bigcup_{t \in T} (q_a, t)))^* \quad (5)$$

$$\cup \bigcup_{c_1, c_2, c_3 \in \Delta} \Delta^* \cdot c_1 c_2 c_3 \cdot \Delta^{2^n - 1} \cdot (\Delta \setminus N(c_1, c_2, c_3)) \cdot \Delta^*, \quad (6)$$

where $q_o, q_a \in Q$, $x_1, \ldots, x_n \in T$, and $N(c_1, c_2, c_3) \subseteq \Delta$.

We now recall some results in [8] for subsequent proof.

Lemma 2. *The formulas* (1)–(6) *can be translated to total NFAs $G_1, \ldots, G_m$, where every G_i is either an NFA or a modular NFA.*[1] *This translation can be performed in polynomial time, with $m = |\Delta|^3 + 5$, such that $L(E) = \bigcup_{i=1}^m P(L_m(G_i))$, where $P \colon \Sigma^* \to \Delta^*$ is a projection with Σ being the alphabet of $\|_{i=1}^m G_i$ and $\Delta \subseteq \Sigma$.* ■

Theorem 2. *Deciding weak prognosability of a live and convergent modular NFA is EXPSPACE-complete.*

Proof. Let $G = \|_{i=1}^n G_i$, and let k be the maximum number of states over G_i. Since the states of $\mathrm{Obs}(G)$ are subsets of n-tuples, each state is of size up to k^n. Using the nondeterministic search (cf. the proof of Theorem 1) gives that the problem can be decided in EXPSPACE.

To prove EXPSPACE-hardness, we leverage Lemma 2. Following the construction of [8] with a slight modification, we add a new observable event $\diamond$ to modify every G_i by adding three new states q_{s_i}, q'_{s_i}, and q_{f_i} such that if a word w is accepted by $P(G_i)$, then q_{s_i} and q_{f_i} are both reachable by $w\diamond$ in $P(G_i)$, whereas if w is not accepted by $P(G_i)$, then q_{s_i} is reachable by $w\diamond$ in $P(G_i)$ but q_{f_i} is not.

Formally, let q_{s_i}, q'_{s_i}, and q_{f_i}, for $i = 1, \ldots, m$, be new non-accepting states. To q'_{s_i}, we add self-loops under the events of $\Delta \cup \{\diamond\}$, where $\diamond$ is a new observable event. For every $i = 1, \ldots, m$, if G_i is an NFA, we modify G_i by adding a transition from every state to state q_{s_i} under the

[1] Here, a modular NFA refers to a parallel composition of NFAs, not the result of the parallel composition.

event $\diamond$, and from every marked state to state q_{f_i}. In addition, we add new unobservable events e_f and $\square_j$, $j = 1, \ldots, m$, where the event e_f is the single fault event, and transitions from q_{s_i} to q'_{s_i} under e_f and $\square_j$, $j = 1, \ldots, m$, $j \neq i$, and from q_{f_i} to q'_{s_i} under $\square_i$, see Fig. 4 for an illustration.

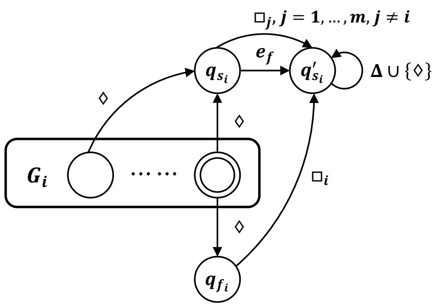

Fig. 4. Modification of the NFA G_i.

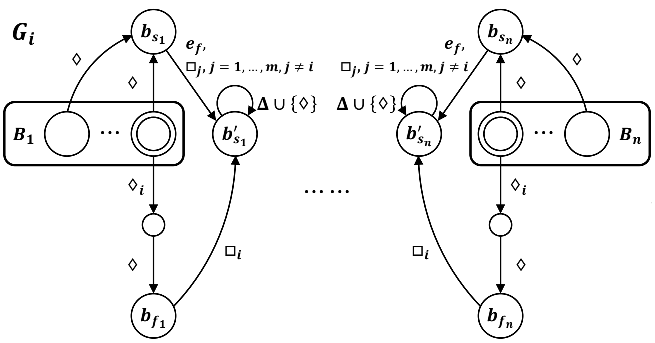

Fig. 5. Modification of the modular NFA G_i.

If G_i is a modular DES $\{B_1, \ldots, B_n\}$, the construction is a bit more complicated. In this case, we add a new unobservable event $\diamond_i$ and, for every B_k, we add three new states b_{s_k}, b'_{s_k}, and b_{f_k} to B_k. Moreover, we add a transition under $\diamond$ from every state of B_k to b_{s_k}, also for every marked state t_k of B_k, we add the transitions $(t_k, \diamond_i, t_k')$ and $(t_k', \diamond, b_{f_k})$, where t_k' is a new state added to B_k. Finally, from b_{s_k} to b'_{s_k}, we add the transitions under e_f and $\square_j$ for $j = 1, \ldots, m$ and $j \neq i$, and from b_{f_k} to b'_{s_k}, add the transition under $\square_i$. See Fig. 5 for an illustration. We define the required states $q_{s_i} = (b_{s_1}, \ldots, b_{s_n})$, $q'_{s_i} = (b'_{s_1}, \ldots, b'_{s_n})$, and $q_{f_i} = (b_{f_1}, \ldots, b_{f_n})$.

It remains to show that $L(E) \neq \Delta^*$ if and only if $\|_{i=1}^m G_i$ is weakly prognosable with respect to P and $\{e_f\}$.

If $L(E) = \Delta^*$, then for every word $w \in \Delta^*$, there exists i such that w is accepted by $P(G_i)$. Then, after observing $w\diamond$, the modular system $\|_{i=1}^m G_i$ can be in at least two different states $\hat{q}_s = (q_{s_1}, q_{s_2}, \ldots, q_{s_m})$ and $\hat{q} = (x_1, x_2, \ldots, x_m)$, where x_j is either q_{f_j} or q_{s_j}, for $j \in \{1, 2, \ldots, m\}$, and there is $i \in \{1, \ldots, m\}$ such that $x_i = q_{f_i}$. Consequently, we can always replace the transition under e_f with a transition under $\square_i$, for some i such that $P(G_i)$ accepts w. Therefore, we cannot predict the occurrence of e_f, and hence the modular system $\|_{i=1}^m G_i$ is not weakly prognosable.

Conversely, if $L(E) \neq \Delta^*$, then there exists a word $w \in \Delta^*$ such that $w \notin L(E) = \bigcup_{i=1}^m P(L_m(G_i))$. We take $K = 1$ and the prefix $w\diamond$ of the faulty word $w\diamond e_f$. After observing $w\diamond$, the modular system $\|_{i=1}^m G_i$ can be in one of the states $\hat{q}_s = (q_{s_1}, q_{s_2}, \ldots, q_{s_m})$ or $(q'_{s_1}, q'_{s_2}, \ldots, q'_{s_m})$, but not in the state $\hat{q} = (x_1, x_2, \ldots, x_m)$, where x_j is either q_{f_j} or q_{s_j}, for $j \in \{1, 2, \ldots, m\}$, and there is $i \in \{1, \ldots, m\}$ such that $x_i = q_{f_i}$. Hence, no transition under $\square_j$ is applicable, and we predict that the fault e_f will occur in the next step. Then the modular system $\|_{i=1}^m G_i$ is weakly prognosable. $\square$

As pointed out by [8], if all unobservable events are private, then projection and parallel composition commute, that is, $P(L_m(\|_{i=1}^m G_i)) = \|_{i=1}^m P(L_m(G_i))$. This implies that $\mathrm{Diag}(\|_{i=1}^m G_i) = \|_{i=1}^m \mathrm{Diag}(G_i)$, allowing us to compute the parallel composition of local diagnosers instead of the diagnoser for the exponentially larger monolithic system. As a consequence of Theorem 2, the complexity of deciding weak modular prognosability becomes PSPACE-complete.

Theorem 3. *Given a live and convergent modular NFA $\|_{i=1}^n G_i$ and a projection $P\colon \Sigma^* \to \Sigma_o^*$ such that all events shared by every pair of NFAs belong to Σ_o, to decide weak modular prognosability is a PSPACE-complete problem.* ■

IV. Deciding Weak Prognosability for LPN

Since Petri net languages (represented by LPNs) possess greater language expressiveness than regular languages (represented by NFAs), there is a general expectation to extend the results obtained for NFAs to LPNs.

Let $N = (P, T, \mathcal{F})$ be a PN, let $G = (N, M_0, \Sigma, \ell)$ be an LPN, and let $T_f \subseteq T$ be the set of fault transitions. We call the transition sequence from $T^*T_fT^*$ *faulty*, and the others *correct*. We denote by $\Psi_G(T_f) = L(N, M_0) \cap T^*T_f$ the set of all sequences in G ending with a fault transition from T_f. For a sequence $\sigma = t_1 \cdots t_n \in T^*$, we write $T_f \in \sigma$ to denote that a fault transition occurs in σ, that is, there is $i \in \{1, \ldots, n\}$ such that $t_i \in T_f$.

We first provide the definition of weak prognosability within LPNs and then discuss its decidability.

Definition 4 (Weak Prognosability of LPNs)**.** *A live and convergent LPN $G = (N, M_0, \Sigma, \ell)$ is* weakly prognosable *with respect to a set of transitions T_f if*

$$(\exists s \in \Psi_G(T_f))(\exists s' \in \overline{s}\colon T_f \notin s')$$
$$(\forall \sigma' \in L(N, M_0)\colon \ell(\sigma') = \ell(s') \wedge T_f \notin \sigma')$$
$$(\exists K \in \mathbb{N})(\forall \sigma \in L(N, M_0)/\sigma')\left[(|\sigma| \geq K) \Rightarrow (T_f \in \sigma)\right].$$

Now, we show that deciding weak prognosability for unbounded LPNs is undecidable.

Theorem 4. *The problem of verifying weak prognosability for live and convergent unbounded LPNs is undecidable.*

Proof. Consider two LPNs G_1 and G_2, both devoid of unobservable transitions. We reduce the language inclusion problem [26] to a weak prognosability verification problem.

Given two LPNs G_1 and G_2, we construct an LPN G consisting of G_1 and G_2 together with six new places, p_0 up to p_5, six new transitions, t_1 up to t_6, labeled by a new label a, and a new fault transition t_f. For $i = 1, 2$, the firing of t_i initializes G_i, and a self-loop connects p_i to every transition of G_i. The rest of G is defined as depicted in Fig. 6.

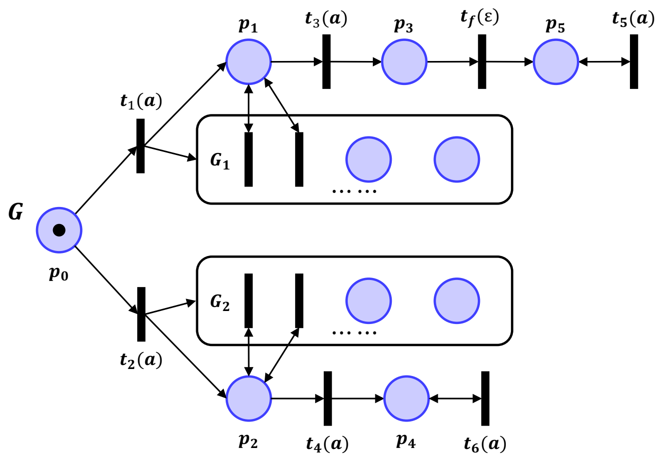

Fig. 6. Sketch of the reduction in the proof of Theorem 4; the notation $t_i(a)$ stands for the transition t_i labeled by a.

We show that $L(G_1) \subseteq L(G_2)$ if and only if G is not weakly prognosable. If $L(G_1) \subseteq L(G_2)$, then every word of the form $awa\varepsilon a^*$ with $w \in L(G_1)$, corresponding to the transition sequence $t_1\sigma t_3 t_f t_5^*$ with σ in G_1 and $\ell(\sigma) = w$, can be simulated using the corresponding part of G_2. Therefore, before fault t_f occurs, no observations leads to fault prediction, resulting in G not being weakly prognosable.

If $L(G_1) \nsubseteq L(G_2)$, then there is a word $w \in L(G_1)$ such that $w \notin L(G_2)$. Let $s = t_1\sigma t_3 t_f$ and $s' = t_1\sigma t_3$ with $\ell(\sigma) = w$. For every σ' such that $\ell(\sigma') = \ell(s') = awa$, the fault event t_f will definitely occur for every continuation σ with $|\sigma| \geq 1$. Thus, the system G is weakly prognosable. $\square$

Remark 1. *Our definition follows that of dynamic diagnosability, where the number of steps K depends on the specific word. An alternative definition is uniform, where the number of steps K is unique for all words [14]. Our proof shows that weak prognosability is undecidable for both alternatives.*

V. Conclusion

In this paper, we introduced weak prognosability for DESs modeled by NFAs and LPNs. Weak prognosability captures the partial predictability of complex systems in which certain faults may be intrinsically unpredictable due to limited sensor coverage. We showed that deciding weak prognosability for NFAs is PSPACE-complete, while for modular NFAs, it is EXPSPACE-complete. However, if all unobservable events are private, the complexity reduces to PSPACE-complete. In addition, we demonstrated that deciding weak prognosability for unbounded LPNs is undecidable. As a direction for future work, hybrid control strategies could be developed for weakly prognosable systems, in which preemptive control is applied along prognosable trajectories to prevent faults from occurrence, while reactive recovery mechanisms are employed for non-prognosable behaviors.

References

[1] S. Miao, A. Lai, J. Komenda, and S. Lahaye, “Decentralized fault diagnosis for constant-time automata,” *IEEE Control Systems Letters*, vol. 8, pp. 3392–3397, 2025.

[2] S. Miao, J. Komenda, and A. Lai, “Active diagnosis of time-interval automata: Time perspectives,” *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 11239–11249, 2025.

[3] S. Genc and S. Lafortune, “Predictability of event occurrences in partially-observed discrete-event systems,” *Automatica*, vol. 45, no. 2, pp. 301–311, 2009.

[4] T.-S. Yoo and S. Lafortune, “Polynomial-time verification of diagnosability of partially observed discrete-event systems,” *IEEE Transactions on Automatic Control*, vol. 47, no. 9, pp. 1491–1495, 2002.

[5] S. Haar, S. Haddad, S. Schwoon, and L. Ye, “Active prediction for discrete event systems,” in *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, 2020.

[6] S. Haar, S. Haddad, T. Melliti, and S. Schwoon, “Optimal constructions for active diagnosis,” *Journal of Computer and System Sciences*, vol. 83, no. 1, pp. 101–120, 2017.

[7] X. Yin and S. Lafortune, “Verification complexity of a class of observational properties for modular discrete events systems,” *Automatica*, vol. 83, pp. 199–205, 2017.

[8] T. Masopust and X. Yin, “Complexity of detectability, opacity and A-diagnosability for modular discrete event systems,” *Automatica*, vol. 101, pp. 290–295, 2019.

[9] S. Miao, B. Cui, Y. Ji, and X. Yin, “Protect your knowledge: Epistemic property enforcement of discrete event systems with asymmetric information,” *IEEE Control Sys. Letters*, vol. 9, pp. 1832–1837, 2025.

[10] N. Bertrand, S. Haddad, and E. Lefaucheux, “Foundation of diagnosis and predictability in probabilistic systems,” in *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, vol. 29, pp. 417–429, 2014.

[11] J. Chen and R. Kumar, “Stochastic failure prognosis of discrete event systems,” *IEEE Transactions on Automatic Control*, vol. 67, no. 10, pp. 5487–5492, 2021.

[12] X. Yin, J. Chen, Z. Li, and S. Li, “Robust fault diagnosis of stochastic discrete event systems,” *IEEE Transactions on Automatic Control*, vol. 64, no. 10, pp. 4237–4244, 2019.

[13] J. Chen, “A probabilistic test for a-diagnosability of stochastic discrete-event systems with guaranteed error bound,” *IEEE Control Systems Letters*, vol. 7, pp. 2833–2838, 2023.

[14] B. Bérard, S. Haar, S. Schmitz, and S. Schwoon, “The complexity of diagnosability and opacity verification for Petri nets,” *Fundamenta Informaticae*, vol. 161, no. 4, pp. 317–349, 2018.

[15] S. Miao, P. Jančar, J. Komenda, T. Masopust, and A. Lai, “Diagnosability verification for automata and Petri nets: Can we do better?,” *Automatica*, 2026. accepted.

[16] D. Lefebvre, “Fault diagnosis and prognosis with partially observed Petri nets,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 10, pp. 1413–1424, 2014.

[17] X. Yin, “Verification of prognosability for labeled Petri nets,” *IEEE Trans. on Automatic Control*, vol. 63, no. 6, pp. 1828–1834, 2017.

[18] D. You, S. Wang, and C. Seatzu, “Verification of fault-predictability in labeled Petri nets using predictor graphs,” *IEEE Transactions on Automatic Control*, vol. 64, no. 10, pp. 4353–4360, 2019.

[19] B. I. de Freitas, A. Toguyeni, and J. C. Basilio, “Online fault prognosis of labeled petri nets,” *IEEE Control Systems Letters*, 2025.

[20] L. Cao, S. Shu, F. Lin, Q. Chen, and C. Liu, “Weak diagnosability of discrete-event systems,” *IEEE Transactions on Control of Network Systems*, vol. 9, no. 1, pp. 184–196, 2021.

[21] M. Sipser, *Introduction to the Theory of Computation*. Cengage Learning, 2012.

[22] C. G. Cassandras and S. Lafortune, *Introduction to discrete event systems*. Springer, 2021.

[23] J. L. Peterson, *Petri net theory and the modeling of systems*. Prentice Hall PTR, 1981.

[24] J. Kao, N. Rampersad, and J. O. Shallit, “On NFAs where all states are final, initial, or both,” *Theoretical Computer Science*, vol. 410, no. 47-49, pp. 5010–5021, 2009.

[25] A. R. Meyer and L. J. Stockmeyer, “The equivalence problem for regular expressions with squaring requires exponential space,” in *13th Annual Symposium on Switching and Automata Theory*, vol. 72, pp. 125–129, 1972.

[26] J. Esparza, “Decidability and complexity of Petri net problems—an introduction,” in *Advanced course on Petri nets*, pp. 374–428, 1996.