Ran, N., Li, T., He, Z., Seatzu, C.

Codiagnosability Enforcement in Labeled Petri Nets

IEEE Transactions on Automatic Control

Volume 68, Fascicolo 4, Pagg. 2436–2443

Codiagnosability Enforcement in Labeled Petri Nets

Ning Ran, Tingting Li, Zhou He, and Carla Seatzu

Abstract

This work aims to enforce *codiagnosability* in labeled Petri nets that are monitored by a series of sites. A labeled Petri net is codiagnosable with respect to a certain fault, if the occurrence of such a fault could be detected by at least one of the sites. We assume that codiagnosability is imposed to a non-codiagnosable system by appropriately positioning additional sensors. In particular, the goal is that of minimizing the cost of the new sensors. The enumeration of the whole state space is avoided thanks to the notions of basis markings and minimal explanations. An automaton, called Unfolded Verifier, is introduced to verify codiagnosability. Finally, the set of optimal labeling functions is obtained solving an integer nonlinear programming problem.

Index Terms

Discrete event systems, fault diagnosis, basis markings, codiagnosability enforcement

I. Introduction

Diagnosability is an important property that must hold when a fault diagnosis method applies in discrete event systems (DESs). Diagnosability analysis consists in verifying a priori if failures can be detected in finite steps. A lot of researchers have proposed a series of methods both in a centralized setting [1]–[3] and in a decentralized setting [4]–[11].

In a Petri net model, an observable event corresponds to a transition whose occurrence may be monitored by a sensor. In particular, a transition is said to be *observable* if it corresponds to an event to which a sensor is associated in the physical system; *unobservable* otherwise. We can analyse diagnosability using the information conveyed by the set of sensors. Sensor configuration may be appropriately modified to make diagnosable a system that is not diagnosable. In this paper we show how this can be done in a decentralized setting with the aim of minimizing the total cost associated with the reconfiguration.

Cabasino *et al.* [12] propose a sensor reconfiguration approach to make a Petri net system diagnosable while optimizing a certain objective function that typically takes into account the cost of sensors. They analyze diagnosability based on the unfolded reachability/coverability graph of the *Verifier Net (VN)*. However, since the scale of reachability/coverability graph could be quite large, the implementation of the method may reveal unfeasible in practical cases.

The notion of basis marking was first proposed in [13] for estimating the markings of Petri nets with silent transitions. It provides a compact way to describe the set of reachable markings consistent with the observations. Then this method was efficiently extended to study some problems related to partial observation, such as diagnosis [14]–[18], prognosis [19], detectability [20]–[22], etc.

To avoid exhaustive enumeration of the whole state space, [17] improves the computational efficiency of the diagnosability enforcement approach in [12] by using the notion of basis marking. In more detail, a particular automaton, called Unfolded Verifier, is defined to perform diagnosability verification of labeled Petri net systems. The paths in the Unfolded Verifier responsible for undiagnosability are identified, then some transitions in such paths are relabeled using appropriate rules proposed in [12]. Finally, given an objective function, an optimal labeling function that guarantees diagnosability is determined.

This work was supported in part by the National Natural Science Foundation of China under Grant 61903119 and Grant 61803246, in part by the Natural Science Foundation of Hebei Province under Grant A2020201021, in part by the Hebei Province Foundation for Returned Overseas Chinese Scholars under Grant C20190319, in part by the Foundation of Hebei Education Department under Grant BJ2021008, in part by the China Postdoctoral Science Foundation under Grant 2019M663608.

N. Ran and T. Li are with Laboratory of Energy-Saving Technology, College of Electronic Information Engineering, Hebei University, Baoding 071002, China. ranning87@hotmail.com

Z. He is with College of Mechanical & Electrical Engineering, Shaanxi University of Science & Technology, Xi'an 710021, China. hzakyhr@gmail.com

C. Seatzu is with DIEE, University of Cagliari, Cagliari 09124, Italy. carla.seatzu@unica.it

This work extends the approach in [17] to a decentralized setting, i.e., enforce *codiagnosability* in Petri nets using optimal sensor selection. More precisely, some sites monitor the evolution of the system and a fault is detected if and only if at least one site is able to do that. If such is the case for any possible occurrence of the fault, the system is said codiagnosable with respect to such a fault. The notion of Unfolded Verifier and some results are extended to a decentralized setting. We show how to find the paths in the Unfolded Verifier that are responsible of the uncodiagnosability of the system. Then, we provide a series of rules for relabeling some transitions in such paths. Finally, given an objective function that we want to minimize, we find the optimal labeling function solving an integer nonlinear programming (INLP) problem.

II. Preliminaries

A Petri net (PN) is a 4-tuple $N = (P, T, F, W)$, where P is the set of places and T is the set of transitions; $F \subseteq (P \times T) \cup (T \times P)$ is the set of arcs; W is a mapping that assigns a non-negative integer weight to an arc: $W(x, y) > 0$ iff $(x, y) \in F$, and $W(x, y) = 0$ otherwise, where $x, y \in P \cup T$. The incidence matrix $[N]$ is a $|P| \times |T|$ integer matrix with $[N](p, t) = W(t, p) - W(p, t)$.

A marking m is a mapping from P to $\mathbb{N} = \{0, 1, 2, ...\}$: $m(p)$ denotes the number of tokens in p. A PN system with an initial marking m_0 is denoted by (N, m_0).

A transition t is enabled at m if for each place p in the preset of t: $m(p) \geq W(p, t)$, and we write $m[t\rangle$. Firing transition t leads to marking m' where $\forall p \in P, m'(p) = m(p) + [N](p, t)$, and we write $m[t\rangle m'$. The notation $m[\sigma\rangle$ is used to denote that transition sequence $\sigma = t_1 t_2 ... t_k$ is enabled at m. Marking m'' is reachable from m if there exists a transition sequence σ such that $m[\sigma\rangle m''$. The reachability set of (N, m), denoted by $R(N, m)$, includes all markings that are reacheable from m. The notations $|\sigma|$ and $\pi(\sigma)$ denote the length of σ and the Parikh vector of σ, respectively. The set of sequences enabled at m_0 is denoted by $L(N, m_0)$. The notation $t \in \sigma$ denotes that σ contains t, and $T' \cap \sigma \neq \emptyset$ means that there exists at least one transition in T' contained in σ.

A PN is: *bounded* if $\forall p \in P$, $\forall m \in R(N, m_0)$, $m(p) \leq k$, where k is a positive number; *deadlock-free* if $\forall m \in R(N, m_0)$, $\exists t \in T$ $m[t\rangle$; *acyclic* if it has no directed circuits.

Let $T' \subseteq T$ be a subset of transitions, the new PN $N' = (P, T', F', W)$ is called the T'-induced subnet of N, where F' is the restriction of F to $(P \times T') \cup (T' \times P)$.

A labeled PN system is a triple $(N, m_0, \mathcal{L})$, where $\mathcal{L}$ is a labeling function $\mathcal{L} : T \to A \cup \{\varepsilon\}$, A is the alphabet and ε is the empty string. A transition associated with ε (a symbol in A) by $\mathcal{L}$ is said to be unobservable (observable). The set of unobservable (observable) transitions is denoted by T_u (T_o); $[N]_u$ ($[N]_o$) denotes the restriction of $[N]$ to T_u (T_o).

We extend $\mathcal{L}$ as follows: i) $\mathcal{L}(t) \in A$, if $t \in T_o$; ii) $\mathcal{L}(t) = \varepsilon$, if $t = \varepsilon$ or $t \in T_u$; iii) $\mathcal{L}(\sigma t) = \mathcal{L}(\sigma)\mathcal{L}(t)$, if $\sigma \in T^*$ and $t \in T$.

Given a label sequence $w \in A^*$, we denote by $\mathcal{L}^{-1}(w)$ the set of transition sequences that are consistent with w, namely $\mathcal{L}^{-1}(w) = \{\sigma \in L(N, m_0) \mid \mathcal{L}(\sigma) = w\}$. Some transition sequences are *indistinguishable* if they have the same labels; *distinguishable* otherwise.

Let $B \subseteq T^*$ be a language, the post-language of B after s is denoted by B/s, i.e., $B/s = \{s' \in T^* \mid ss' \in B\}$.

III. Problem Formulation

The set T_u of unobservable transitions consists of the set of regular transitions T_{reg} and the set of faults T_f, i.e., $T_u = T_{reg} \cup T_f$. The set T_{reg} consists of $T_{r,o}$ and $T_{r,uo}$, i.e., $T_{reg} = T_{r,o} \cup T_{r,uo}$, where $T_{r,o}$ ($T_{r,uo}$) denotes the set of regular transitions to which it is possible (not possible) to associate a sensor. The set T_f includes r different fault classes T_f^1, T_f^2, ..., T_f^r.

A. Codiagnosability of labeled PNs

A set of sites $\mathcal{J} = \{1, 2, ..., \nu\}$ monitors the system with their own masks. We assume that each transition in T_o is observable by at least one site, i.e., $T_o = \bigcup_{j \in \mathcal{J}} T_{o,j}$, where $T_{o,j} \subseteq T_o$ is the set of transitions that are observable by site j. Accordingly, $T_{u,j} = T \setminus T_{o,j}$ denotes the set of transitions that are unobservable by site j. Moreover, we assume that none of the symbols in A is the same as the name of a transition, i.e., $A \cap T \neq \emptyset$.

The alphabet of site j is $A_j \subseteq A$, and

$$\mathcal{L}_j(t) = \begin{cases} \mathcal{L}(t), \text{ if } \mathcal{L}(t) \in A_j \\ \varepsilon, \text{ otherwise} \end{cases} \tag{1}$$

denotes the labeling function of site j.

We denote by $\Psi(T_f^i)$ the set of sequences of $L(N, m_0)$ ending with a fault in T_f^i.

Definition 1 ([16]): Given a system $(N, m_0, \mathcal{L})$ that is deadlock-free after any fault, it is *codiagnosable* wrt T_f^i if
$(\forall s \in \Psi(T_f^i)),\ (\exists K \in \mathbb{N}),\ (\forall \sigma \in L(N, m_0)/s),\ |\sigma| \geq K \Rightarrow$
$(\exists j \in \mathcal{J}),\ (\forall \sigma' \in \mathcal{L}_j^{-1}(\mathcal{L}_j(s\sigma))),\ T_f^i \cap \sigma' \neq \emptyset$
The labeled PN system is *codiagnosable* if it is codiagnosable wrt all fault classes.

In other words, a labeled PN system is codiagnosable wrt T_f^i if at least one site is able to detect any fault in T_f^i in finite steps.

B. Problem definition and relabeling rules

Given a non-codiagnosable system $(N, m_0, \mathcal{L})$ monitored by a set $\mathcal{J} = \{1, 2, ..., \nu\}$ of sites, our objective is to identify ν new labeling functions:

$$\mathcal{L}_{j,new} : T \to A_{j,new} \cup \{\varepsilon\},\ j = 1, 2, \ldots, \nu,$$

such that $(N, m_0, \mathcal{L}_{new})$ is codiagnosable, where $A_{j,new}$ is the new alphabet of site j, and

$$\mathcal{L}_{new}(t) = \begin{cases} \mathcal{L}_{j,new}(t), \text{ if } \exists j \in \mathcal{J} :\ \mathcal{L}_{j,new}(t) \in A_{j,new} \\ \varepsilon, \text{ otherwise} \end{cases}$$

We also denote $A_{new} = \bigcup_{j \in \mathcal{J}} A_{j,new}$.

In simple words, to enforce codiagnosability we select some transitions in the set $T_o \cup T_{r,o}$ to assign new symbols. Without loss of generality, the new symbols are the names of the transitions, namely $A_{j,new} = A_j \cup (T_{o,j} \cup T_{r,o})$. In particular, the transitions in $T_{r,o} \cup T_o$ are relabeled by two rules:

(R1) Given $t \in T_{r,o}$, we either relabel it as

- $\mathcal{L}_{new}(t) = t$ and
$$\mathcal{L}_{j,new}(t) = \begin{cases} t, \text{ if } j \in \bar{\mathcal{J}} \\ \varepsilon, \text{ otherwise} \end{cases} \tag{2}$$
for a certain set of sites $\bar{\mathcal{J}} \subseteq \mathcal{J}$; or
- leave $\mathcal{L}_{new}(t)$ unchanged as $\mathcal{L}_{new}(t) = \varepsilon$ and $\mathcal{L}_{j,new}(t) = \varepsilon$ for all $j \in \mathcal{J}$.

(R2) For a transition $t \in T_o$ such that there exists another transition $t' \in T_o$ that is indistinguishable from t, we either relabel it as

- $\mathcal{L}_{new}(t) = t$ and
$$\mathcal{L}_{j,new}(t) = \begin{cases} t, \text{ if } t \in T_{o,j} \\ \varepsilon, \text{ otherwise} \end{cases} \tag{3}$$
or
- leave $\mathcal{L}_{new}(t)$ unchanged as $\mathcal{L}_{new}(t) = \mathcal{L}(t)$ and
$$\mathcal{L}_{j,new}(t) = \begin{cases} \mathcal{L}(t), \text{ if } t \in T_{o,j} \\ \varepsilon, \text{ otherwise} \end{cases} \tag{4}$$

According to rule R1, transition $t \in T_{r,o}$ may become observable for some sites, assigning to it the new unique label t, i.e., t itself. According to rule R2, to transition $t \in T_o$ it may be assigned the new unique label t. This means that t becomes distinguishable for all those sites for which it was already observable.

In the following discussion, we assume that:

(A1) There is one fault class.

(A2) The PN system is bounded.

(A3) The $T_{u,j}$-induced subnets are acyclic for all $j \in \mathcal{J}$.
(A4) The PN system is deadlock-free after the occurrence of a fault.
(A5) The PN system $(N, m_0, \mathcal{L}_{total})$ is codiagnosable, where $\mathcal{L}_{total}$ is defined as follows:

$$\mathcal{L}_{total}(t) = \begin{cases} t, & \text{if } t \in T_o \cup T_{r,o} \\ \varepsilon, & \text{otherwise.} \end{cases} \tag{5}$$

and for any site $j \in \mathcal{J}$, it holds that

$$\mathcal{L}_{j,total}(t) = \begin{cases} t, & \text{if } t \in T_{o,j} \cup T_{r,o} \\ \varepsilon, & \text{otherwise.} \end{cases} \tag{6}$$

Assumption A1 is made for simplicity, and will be relaxed in Section VII-B. Assumptions A2 and A3 are necessary for using the notion of basis marking [15]. The fourth assumption is typical in the literature on fault diagnosis of DESs and avoids handling the case that a PN system is dead after a fault. The last assumption guarantees that there exists at least one solution to the considered problem.

IV. Extended Basis Reachability Graph

We first recall some preliminaries proposed in [15], [16].

Definition 2 ([15]): Given $m \in R(N, m_0)$ and $t \in T_o$, the set of *explanations* of t at m is denoted by $\Sigma(m,t) = \{\sigma \in T_u^* \mid m[\sigma\rangle m', m'[t\rangle\}$, and the set of *e-vectors* is denoted by $Y(m,t) = \pi(\Sigma(m,t))$.

Definition 3 ([15]): Given $m \in R(N, m_0)$ and $t \in T_o$, the set of *minimal explanations* of t at m is denoted by $\Sigma_{min}(m,t) = \{\sigma \in \Sigma(m,t) \mid \nexists \sigma' \in \Sigma(m,t) : \pi(\sigma') \lneq \pi(\sigma)\}$, and the set of *minimal e-vectors* is denoted by $Y_{min}(m,t) = \pi(\Sigma_{min}(m,t))$.

Definition 4 ([15]): Let w be an observation. The set of pairs ($\sigma_o \in T_o^*$ with $\mathcal{L}(\sigma_o) = w$ and the justification) is denoted by $\hat{\mathcal{J}}(w) = \{(\sigma_o, \sigma_u), \sigma_o \in T_o^*, \mathcal{L}(\sigma_o) = w, \sigma_u \in T_u^* \mid [\exists \sigma \in \mathcal{L}^{-1}(w) : \sigma_o = P_o(\sigma), \sigma_u = P_u(\sigma)] \wedge [\nexists \sigma' \in \mathcal{L}^{-1}(w) : \sigma_o = P_o(\sigma'), \sigma'_u = P_u(\sigma') \wedge \pi(\sigma'_u) \lneq \pi(\sigma_u)]\}$, and the set of pairs ($\sigma_o \in T_o^*$ with $\mathcal{L}(\sigma_o) = w$ and the j-vector) is denoted by $\hat{Y}_{min}(m_0, w) = \{(\sigma_o, y), \sigma_o \in T_o^*, \mathcal{L}(\sigma_o) = w, y \in \mathbb{N}^{|T_u|} \mid \exists (\sigma_o, \sigma_u) \in \hat{\mathcal{J}}(w) : \pi(\sigma_u) = y\}$.

Definition 5 ([15]): Let w be an observation and $\hat{\mathcal{J}}(w)$ be a set of pairs. The set of *basis markings (BM)* of w is denoted by $M_b(w) = \{m \in \mathbb{N}^{|P|} \mid m = m_0 + [N]_u \cdot \pi(\sigma_u) + [N]_o \cdot \pi(\sigma_o), (\sigma_o, \sigma_u) \in \hat{\mathcal{J}}(w)\}$, and $M_b = \bigcup_{w \in A^*} M_b(w)$.

In simple words, a basis marking is a marking that can be reached from the initial marking firing a sequence of transitions that is consistent with the observation and a sequence of unobservable transitions, interleaved with the previous sequence, whose firing is strictly necessary to enable it (in the sense that its firing vector is minimal). The set of basis markings is a subset (usually a strict subset) of the set of reachable markings. Therefore, if the net is bounded, the set of basis markings is finite.

The following definition is inspired by [16].

Definition 6: An *extended basis marking (EBM)* is a BM computed assuming that all transitions in $T_f \cup T_{r,o}$ are observable. We denote by M_e the set of EBMs.

The set of EBMs can be computed by restricting the minimal explanations to $T_{r,uo}$. Hereinafter, we use $Y_{min}^{r,uo}(m,t)$ to denote the set of minimal e-vectors restricted to $T_{r,uo}$. The set $Y_{min}^{r,uo}(m,t)$ can be computed by Algorithm 4.4 in [15].

Example 1: Fig. 1 shows a labeled PN model $(N, m_0, \mathcal{L})$ of a manufacturing process, where transitions model operations that should be executed according a certain order, and places represent the working conditions of certain machines, conveyors, or buffers. With some transitions, sensors are attached, while no sensor is attached to the others. In this model, $T_o = \{t_3, t_7, t_8, t_{11}\}$, $T_u = \{t_1, t_2, t_4 - t_6, t_9, t_{10}\}$, $T_f = \{t_9\}$, $T_{r,o} = \{t_1, t_4\}$, $T_{r,uo} = \{t_2, t_5, t_6, t_{10}\}$ and $m_0 = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T$. Let $\mathcal{L}(t_3) = a$, $\mathcal{L}(t_7) = b$ and $\mathcal{L}(t_8) = \mathcal{L}(t_{11}) = c$. It holds that $\Sigma(m_0, t_8) = \{t_1\}$, while $\Sigma(m_0, t_{11}) = \emptyset$. Given a marking $m = [0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0]^T$, we have $\Sigma(m, t_3) = \{\varepsilon, t_2\}$ and $\Sigma_{min}(m, t_3) = \{\varepsilon\}$. It is $Y(m, t_3) = \{[0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0]^T, [0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0]^T\}$ and $Y_{min}(m, t_3) = \{[0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0]^T\}$.

Given an observation $w = bc$, the sets $\hat{\mathcal{J}}(w) = \{t_7 t_{11}, t_1 t_5 t_6\}$, $\hat{Y}_{min}(m_0, w) = \{t_7 t_{11}, [1\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0]^T\}$ and $M_b(w) = \{[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1]^T\}$.

Finally, the set of EBMs is detailed in Table I.

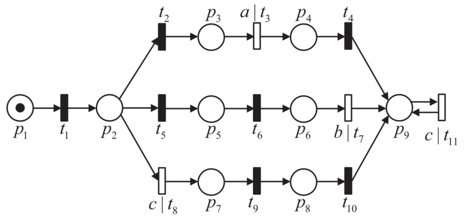


Fig. 1: A labeled PN system $(N, m_0, \mathcal{L})$.

TABLE I: EBMs of the PN system in Example 1.

Node	EBM
m_0	$[1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T$
m_1	$[0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T$
m_2	$[0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0]^T$
m_3	$[0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0]^T$
m_4	$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0]^T$
m_5	$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1]^T$

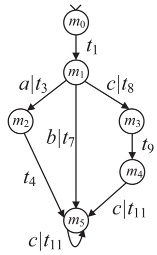


Fig. 2: EBRG G_e.

Let $T' = T \setminus T_f$ and N' be the T'-induced subnet of N. We denote by $(N', m_0, \mathcal{L}')$ the *nonfailure subnet* of $(N, m_0, \mathcal{L})$, where $\mathcal{L}'$ is equal to $\mathcal{L}$ restricted to $T \setminus T_f$. Therefore, $L(N', m_0)$ is the set of transitions sequences of $L(N, m_0)$ containing no faults.

The following two graphs are also inspired by [16].

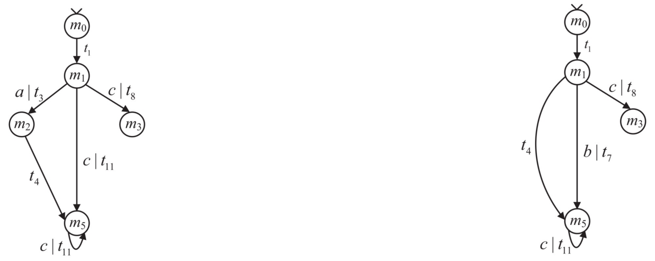
Definition 7: Let $(N', m_0, \mathcal{L}')$ be the nonfailure subnet of $(N, m_0, \mathcal{L})$.

- The *Extended Basis Reachability Graph (EBRG)* is a finite state automaton $G_e = (M_e, E, \Delta, m_0)$, where M_e is the set of EBMs; $E \subseteq (T_o \times A) \cup T_f \cup T_{r,o}$ is the set of event labels; $\Delta \subseteq M_e \times E \times M_e$ is the transition relation; m_0 is the initial state.
- The *nonfailure EBRG wrt site* j, denoted by $G_e^j = (M^j, E^j, \Delta^j, m_0)$, is the EBRG of $(N', m_0, \mathcal{L}')$ constructed by assuming that all transitions in $T_{o,j} \cup T_{r,o}$ $(T' \setminus T_{o,j} \setminus T_{r,o})$ are observable (unobservable).

The EBRG G_e is constructed by Algorithm 1. The graph G_e^j can also be constructed using Algorithm 1, by taking $(N', m_0, \mathcal{L}')$ as the input, by assuming that the set $T_{o,j} \cup T_{r,o}$ includes observable transitions and by restricting minimal explanations to $T' \setminus T_{o,j} \setminus T_{r,o}$.

It should be noted that the above algorithm is different from Algorithm 1 in [16] since here we assume that all transitions in $T_o \cup T_f \cup T_{r,o}$ are observable, thus the minimal explanations are restricted to $T_{r,uo}$.

Example 2: Assume that the system in Example 1 is monitored by two sites with $A_1 = \{a, c\}$ and $A_2 = \{b, c\}$. The EBRG G_e is shown in Fig. 2, the nonfailure EBRGs wrt two sites are reported in Fig. 3a and Fig. 3b, respectively.



(a)

(b)

Fig. 3: G_e^1: nonfailure EBRG wrt site 1; G_e^2: nonfailure EBRG wrt site 2.

Algorithm 1: EBRG construction

```
Input: A labeled PN system $(N, m_0, \mathcal{L})$.
Output: The EBRG $G_e = (M_e, E, \Delta, m_0)$.
Let $M_e = \{m_0\}$, $E = \emptyset$, $\Delta = \emptyset$ and $m_0$ be the initial state.
while states with no tag exist in $M_e$, do
    choose a state $m$ with no tag,
    for all $t \in T_o \cup T_f \cup T_{r,o}$, do
        if $Y_{min}^{r,uo}(m, t) \neq \emptyset$, then
            for all $y \in Y_{min}^{r,uo}(m, t)$, do
                let $m' = m + [N]_{r,uo} \cdot y + [N](\cdot, t)$.
                let $M_e = M_e \cup \{m'\}$.
                if $t \in T_o$ and $\mathcal{L}(t) = e$, then $E = E \cup \{t(e)\}$ and $\Delta = \Delta \cup \{(m, t(e), m')\}$.
                ;
                if $t \in T_f \cup T_{r,o}$, then $E = E \cup \{t\}$ and $\Delta = \Delta \cup \{(m, t, m')\}$.;
            end
        end
    end
    tag state $m$ "old".
end
Remove all tags.
```

V. Unfolded Verifier

In [17] we introduced the Unfolded Verifier (UV) to analyse diagnosability of PNs in a centralized setting, which is similar to other tools previously introduced by other authors, in particular [23]. Here we extend the notion of UV in [17] to a decentralized setting, and propose necessary and sufficient conditions for codiagnosability. For the sake of simplicity, only two local sites are considered in the following discussion. We will consider the case of ν sites in Section VII-B.

The *Unfolded Verifier (UV)* is the finite state automaton constructed by Algorithm 2. In plain words, the UV is constructed as the parallel composition of the EBRG and the nonfailure EBRGs. The expansion of each path is terminated when the new state is either dead or "duplicate". A state $(m, l; m_1; m_2)$ is said to be a *duplicate l-state* if its tag is "duplicate".

Theorem 1: A labeled PN system is codiagnosable iff there exist no duplicate F-states in its UV.

Proof. This result is derived from two facts: 1) a labeled PN is codiagnosable iff its Verifier has no F-cycles (which has been proved in [16]), where an F-cycle is a cycle in which all the nodes are F-states, and 2) the UV can be viewed as the unfolded version of the Verifier. The difference between the construction of the Verifier and the construction of the UV only consists in the way we deal with repeated nodes: the former merges repeated nodes; on the contrary, the latter does not, and assigns "duplicate" to a node if it is the same as another node in the path from the root to it. Hence, the first fact is directly rephrased as the statement of the theorem if we look at the UV. □

Given an automaton G, the notation $m \xrightarrow[G]{\tau} m'$ indicates m' is reached from m in G through a sequence τ.

Definition 8: Given an UV U, a sequence τ in U is said to be an *elementary F-path wrt $\mathcal{L}$* if it starts at the root of U and ends with a duplicate F-state.

Example 3: For the sake of brevity, the UV U of the system in Fig. 1 is not given here. Fig. 4 shows all the elementary F-paths in U, and Table II reports the states in these paths. There are 12 elementary F-paths wrt $\mathcal{L}$ (τ_1 to τ_{12}) and state 14 is a duplicate F-state.

By Definition 8 and Theorem 1, we can directly infer the following result.

Theorem 2: A labeled PN system is codiagnosable iff there exist no elementary F-paths wrt $\mathcal{L}$ in its UV.

Algorithm 2: Construction of the UV

```
Input: $G_e = (M_e, E, \Delta, m_0)$, $G_e^1 = (M^1, E^1, \Delta^1, m_0)$ and $G_e^2 = (M^2, E^2, \Delta^2, m_0)$.
Output: The UV $U = (M^U, E^U, \Delta^U, m_0^U)$.
Let $M^U = \{(m_0, \mathrm{N}; m_0; m_0)\}$, $E^U = \emptyset$, $\Delta^U = \emptyset$, and $m_0^U = (m_0, \mathrm{N}; m_0; m_0)$ be the initial state.
while states with no tag exist, do
    choose a state $(m, l; m_1; m_2)$ with no tag,
    for all $t \in T_o \cup T_f \cup T_{r,o}$ and all $t_1, t_2 \in T_o \cup T_{r,o}$, do
        if $t \in T_{o,1} \cap T_{o,2}$, $(m, t, m') \in \Delta$, $(m_1, t_1, m_1') \in \Delta^1$, $(m_2, t_2, m_2') \in \Delta^2$, $\mathcal{L}_1(t) = \mathcal{L}_1(t_1)$,
          $\mathcal{L}_2(t) = \mathcal{L}_2(t_2)$, then
            $M^U = M^U \cup \{(m', l; m_1'; m_2')\}$, $E^U = E^U \cup \{(t, t_1, t_2)\}$ and
              $\Delta^U = \Delta^U \cup \{(m, l; m_1; m_2), (t, t_1, t_2), (m', l; m_1'; m_2')\}$.
        end
        if $t \in T_f$, $(m, t, m') \in \Delta$, then
            $M^U = M^U \cup \{(m', \mathrm{F}; m_1; m_2)\}$, $E^U = E^U \cup \{(t, \varepsilon, \varepsilon)\}$ and
              $\Delta^U = \Delta^U \cup \{(m, l; m_1; m_2), (t, \varepsilon, \varepsilon), (m', \mathrm{F}; m_1; m_2)\}$.
        end
        if $t \in T_{o,1} \setminus T_{o,2}$, $(m, t, m') \in \Delta$,
          $(m_1, t_1, m_1') \in \Delta^1$, $\mathcal{L}_1(t) = \mathcal{L}_1(t_1)$, then
            $M^U = M^U \cup \{(m', l; m_1'; m_2)\}$, $E^U = E^U \cup \{(t, t_1, \varepsilon)\}$ and
              $\Delta^U = \Delta^U \cup \{(m, l; m_1; m_2), (t, t_1, \varepsilon), (m', l; m_1'; m_2)\}$.
        end
        if $t \in T_{o,2} \setminus T_{o,1}$, $(m, t, m') \in \Delta$,
          $(m_2, t_2, m_2') \in \Delta^2$, $\mathcal{L}_2(t) = \mathcal{L}_2(t_2)$, then
            $M^U = M^U \cup \{(m', l; m_1; m_2')\}$, $E^U = E^U \cup \{(t, \varepsilon, t_2)\}$ and
              $\Delta^U = \Delta^U \cup \{(m, l; m_1; m_2), (t, \varepsilon, t_2), (m', l; m_1; m_2')\}$.
        end
        if $t \in T_{r,o}$, $(m, t, m') \in \Delta$, then
            $M^U = M^U \cup \{(m', l; m_1; m_2)\}$, $E^U = E^U \cup \{(t, \varepsilon, \varepsilon)\}$ and $\Delta^U = \Delta^U \cup \{(m, l;$
              $m_1; m_2), (t, \varepsilon, \varepsilon), (m', l; m_1; m_2)\}$.
        end
        if $t_1 \in T_{r,o}$, $(m_1, t_1, m_1') \in \Delta^1$, then
            $M^U = M^U \cup \{(m, l; m_1'; m_2)\}$, $E^U = E^U \cup \{(\varepsilon, t_1, \varepsilon)\}$ and
              $\Delta^U = \Delta^U \cup \{(m, l; m_1; m_2), (\varepsilon, t_1, \varepsilon), (m, l; m_1'; m_2)\}$.
        end
        if $t_2 \in T_{r,o}$, $(m_2, t_2, m_2') \in \Delta^2$, then
            $M^U = M^U \cup \{(m, l; m_1; m_2')\}$, $E^U = E^U \cup \{(\varepsilon, \varepsilon, t_2)\}$ and
              $\Delta^U = \Delta^U \cup \{(m, l; m_1; m_2), (\varepsilon, \varepsilon, t_2), (m, l; m_1; m_2')\}$.
        end
    end
    if the new state is the same as a state in the path from the initial state to it, then tag it “duplicate".;
    tag state $(m, l; m_1; m_2)$ “old".
end
```

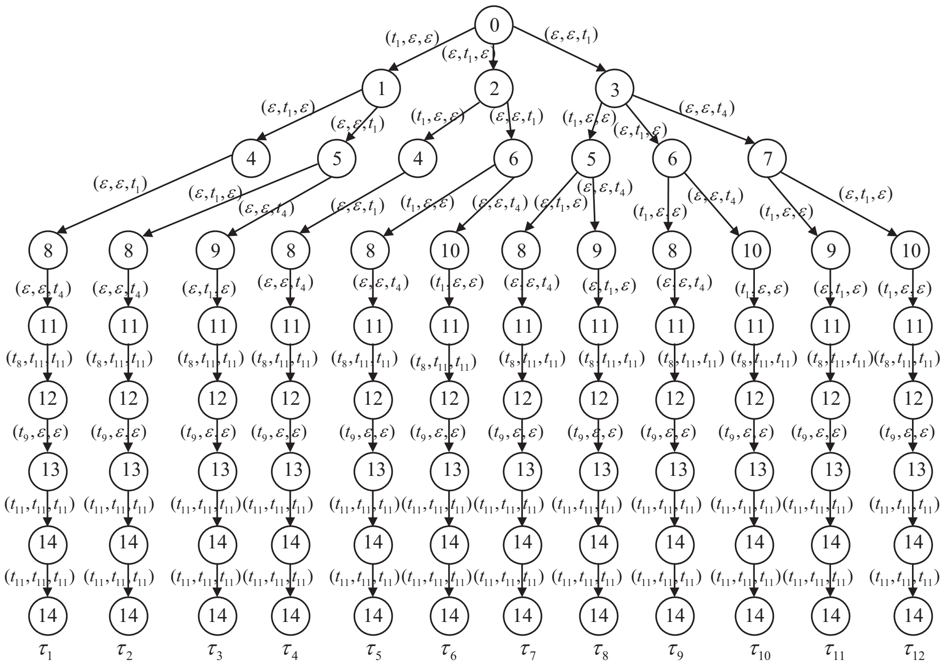

0
$(t_1,\varepsilon,\varepsilon)$
$(\varepsilon,t_1,\varepsilon)$
$(\varepsilon,\varepsilon,t_1)$
1
2
3
$(\varepsilon,t_1,\varepsilon)$
$(\varepsilon,\varepsilon,t_1)$
$(t_1,\varepsilon,\varepsilon)$
$(\varepsilon,\varepsilon,t_1)$
$(t_1,\varepsilon,\varepsilon)$
$(\varepsilon,t_1,\varepsilon)$
$(\varepsilon,\varepsilon,t_4)$
4
5
4
6
5
6
7
$(\varepsilon,\varepsilon,t_1)$
$(\varepsilon,t_1,\varepsilon)$
$(\varepsilon,\varepsilon,t_4)$
$(\varepsilon,\varepsilon,t_1)$
$(t_1,\varepsilon,\varepsilon)$
$(\varepsilon,\varepsilon,t_4)$
$(\varepsilon,t_1,\varepsilon)$
$(\varepsilon,\varepsilon,t_4)$
$(t_1,\varepsilon,\varepsilon)$
$(\varepsilon,\varepsilon,t_4)$
$(t_1,\varepsilon,\varepsilon)$
$(\varepsilon,t_1,\varepsilon)$
8
8
9
8
8
10
8
9
8
10
9
10
$(\varepsilon,\varepsilon,t_4)$
$(\varepsilon,\varepsilon,t_4)$
$(\varepsilon,t_1,\varepsilon)$
$(\varepsilon,\varepsilon,t_4)$
$(\varepsilon,\varepsilon,t_4)$
$(t_1,\varepsilon,\varepsilon)$
$(\varepsilon,\varepsilon,t_4)$
$(\varepsilon,t_1,\varepsilon)$
$(\varepsilon,\varepsilon,t_4)$
$(t_1,\varepsilon,\varepsilon)$
$(\varepsilon,t_1,\varepsilon)$
$(t_1,\varepsilon,\varepsilon)$
11
(t_8,t_{11},t_{11})
12
$(t_9,\varepsilon,\varepsilon)$
13
(t_{11},t_{11},t_{11})
14
(t_{11},t_{11},t_{11})
14
τ_1
τ_2
τ_3
τ_4
τ_5
τ_6
τ_7
τ_8
τ_9
τ_{10}
τ_{11}
τ_{12}

Fig. 4: The elementary F-paths of the labeled PN in Example 1.

TABLE II: The states associated with the nodes in Fig. 4.

Node	States
0	$(m_0, \text{N}; m_0; m_0)$
1	$(m_1, \text{N}; m_0; m_0)$
2	$(m_0, \text{N}; m_1; m_0)$
3	$(m_0, \text{N}; m_0; m_1)$
4	$(m_1, \text{N}; m_1; m_0)$
5	$(m_1, \text{N}; m_0; m_1)$
6	$(m_0, \text{N}; m_1; m_1)$
7	$m_0, \text{N}; m_0; m_5)$
8	$(m_1, \text{N}; m_1; m_1)$
9	$(m_1, \text{N}; m_0; m_5)$
10	$(m_0, \text{N}; m_1; m_5)$
11	$(m_1, \text{N}; m_1; m_5)$
12	$(m_3, \text{N}; m_5; m_5)$
13	$(m_4, \text{F}; m_5; m_5)$
14	$(m_5, \text{F}; m_5; m_5)$

VI. Relabeling of Elementary F-Paths

In this section, we show how to *eliminate* the elementary F-paths in a UV by relabeling some transitions in such paths.

A. Relabeling options and conditions on transitions in $T_{r,o}$

The arcs of the UV are triples $(\gamma_i, \gamma_j, \gamma_k)$, where

i) γ_i denotes a transition in G_e or ε;
ii) γ_j (γ_k) denotes a transition in G_e^1 or ε (G_e^2 or ε).

Based on rules R1 and R2, we consider the *relabeling options* for all the possible triples $(\gamma_i, \gamma_j, \gamma_k)$ in each elementary F-path:

(LO1) $(\gamma_i, \gamma_j, \gamma_k) = (t_i, t_j, t_k)$, where $t_i, t_j, t_k \in T_o$. If $i = j = k$, we do nothing. Otherwise, we may assign to t_i a new label $\mathcal{L}_{new}(t_i) = \mathcal{L}_{1,new}(t_i) = \mathcal{L}_{2,new}(t_i) = t_i$, or/and to t_j a new label $\mathcal{L}_{new}(t_j) = \mathcal{L}_{1,new}(t_j) = \mathcal{L}_{2,new}(t_j) = t_j$, or/and to t_k a new label $\mathcal{L}_{new}(t_k) = \mathcal{L}_{1,new}(t_k) = \mathcal{L}_{2,new}(t_k) = t_k$.

(LO2) $(\gamma_i, \gamma_j, \gamma_k) = (t_i, t_j, \varepsilon)$, where $t_i, t_j \in T_o$. If $i = j$, we do nothing. Otherwise, we may assign to t_i a new label $\mathcal{L}_{new}(t_i) = \mathcal{L}_{1,new}(t_i) = t_i$, or/and to t_j a new label $\mathcal{L}_{new}(t_i) = \mathcal{L}_{1,new}(t_j) = t_j$.

(LO3) $(\gamma_i, \gamma_j, \gamma_k) = (t_i, \varepsilon, t_k)$, where $t_i, t_k \in T_o$. If $i = k$, we do nothing. Otherwise, we may assign to t_i a new label $\mathcal{L}_{new}(t_i) = \mathcal{L}_{2,new}(t_i) = t_i$, or/and to t_k a new label $\mathcal{L}_{new}(t_k) = \mathcal{L}_{2,new}(t_k) = t_k$.

(LO4) $(\gamma_i, \gamma_j, \gamma_k) = (t, \varepsilon, \varepsilon)$, where $t \in T_{r,o}$. We may make t observable for site 1 or/and site 2, i.e., $\mathcal{L}_{1,new}(t) = t$ or/and $\mathcal{L}_{2,new}(t) = t$.

(LO5) $(\gamma_i, \gamma_j, \gamma_k) = (\varepsilon, t, \varepsilon)$, where $t \in T_{r,o}$. We may make t uniquely observable for site 1 or observable for both sites, i.e., $\mathcal{L}_{1,new}(t) = t$, $\mathcal{L}_{2,new}(t) = \varepsilon$ or $\mathcal{L}_{1,new}(t) = \mathcal{L}_{2,new}(t) = t$.

(LO6) $(\gamma_i, \gamma_j, \gamma_k) = (\varepsilon, \varepsilon, t)$, where $t \in T_{r,o}$. We may make t uniquely observable for site 2 or observable for both sites, i.e., $\mathcal{L}_{1,new}(t) = \varepsilon$, $\mathcal{L}_{2,new}(t) = t$ or $\mathcal{L}_{1,new}(t) = \mathcal{L}_{2,new}(t) = t$.

(LO7) $(\gamma_i, \gamma_j, \gamma_k) = (t, \varepsilon, \varepsilon)$, where $t \in T_f$. We do nothing since faults cannot be captured by any sensor.

Now, given an elementary F-path in the UV, at least one transition $t \in T_o \cup T_{r,o}$ is selected to be relabeled according to relabeling options LO1 to LO6. In addition, a transition $t \in T_{r,o}$ could be handled provided that it satisfies the following 5 conditions:

(C1) transition $t \in T_{r,o}$ should not be uniquely observable for site 1 if it *only* appears in *consecutive triples* as follows:
$$(t, \varepsilon, \varepsilon)(\varepsilon, t, \varepsilon) \text{ or } (\varepsilon, t, \varepsilon)(t, \varepsilon, \varepsilon).$$

(C2) transition $t \in T_{r,o}$ should not be uniquely observable for site 2 if it *only* appears in *consecutive triples* as follows:
$$(t, \varepsilon, \varepsilon)(\varepsilon, \varepsilon, t) \text{ or } (\varepsilon, \varepsilon, t)(t, \varepsilon, \varepsilon).$$

(C3) transition $t \in T_{r,o}$ should not be uniquely observable for site 1 and should not be observable for both sites if it *only* appears in *consecutive triples* as follows:
$$(t, \varepsilon, \varepsilon)(\varepsilon, t, \varepsilon)(\varepsilon, \varepsilon, t) \text{ or } (\varepsilon, \varepsilon, t)(\varepsilon, t, \varepsilon)(t, \varepsilon, \varepsilon).$$

(C4) transition $t \in T_{r,o}$ should not be uniquely observable for site 2 and should not be observable for both sites if it *only* appears in *consecutive triples* as follows:
$$(t, \varepsilon, \varepsilon)(\varepsilon, \varepsilon, t)(\varepsilon, t, \varepsilon) \text{ or } (\varepsilon, t, \varepsilon)(\varepsilon, \varepsilon, t)(t, \varepsilon, \varepsilon).$$

(C5) transition $t \in T_{r,o}$ should not be observable for any site if it *only* appears in *consecutive triples* as follows:
$$(\varepsilon, t, \varepsilon)(t, \varepsilon, \varepsilon)(\varepsilon, \varepsilon, t) \text{ or } (\varepsilon, \varepsilon, t)(t, \varepsilon, \varepsilon)(\varepsilon, t, \varepsilon).$$

Here we explain condition C1. If $t \in T_{r,o}$ only appears in consecutive triples $(t, \varepsilon, \varepsilon)(\varepsilon, t, \varepsilon)$ or $(\varepsilon, t, \varepsilon)(t, \varepsilon, \varepsilon)$, and is uniquely observable for site 1, then the elementary F-path may not be eliminated since the consecutive triples are replaced by (t, t, ε) under the new labeling function, and the PN system remains non-codiagnosable. Hence, condition C1 is necessary. Condition C2 is necessary for the same reason.

Condition C3 is explained as follows. Assume that $t \in T_{r,o}$ only appears in consecutive triples $(t, \varepsilon, \varepsilon)(\varepsilon, t, \varepsilon)(\varepsilon, \varepsilon, t)$ or $(\varepsilon, \varepsilon, t)(\varepsilon, t, \varepsilon)(t, \varepsilon, \varepsilon)$.

- If t is uniquely observable for site 1, then the elementary F-path may not be eliminated since the consecutive triples are replaced by $(t, t, \varepsilon)(\varepsilon, \varepsilon, t)$ or $(\varepsilon, \varepsilon, t)(t, t, \varepsilon)$ under the new labeling function, and the PN system remains non-codiagnosable.
- If t is observable for both sites, then the elementary F-path may not be eliminated since the consecutive triples are replaced by (t, t, t) under the new labeling function, and the PN system remains non-codiagnosable.

Hence, condition C3 is necessary. Condition C4 is necessary for the same reason.

Condition C5 is also necessary. Assume that $t \in T_{r,o}$ only appears in consecutive triples $(\varepsilon, t, \varepsilon)(t, \varepsilon, \varepsilon)(\varepsilon, \varepsilon, t)$ or $(\varepsilon, \varepsilon, t)(t, \varepsilon, \varepsilon)(\varepsilon, t, \varepsilon)$.

- If t is uniquely observable for site 1, then the elementary F-path may not be eliminated since the triples are replaced by $(t, t, \varepsilon)(\varepsilon, \varepsilon, t)$ or $(\varepsilon, \varepsilon, t)(t, t, \varepsilon)$ under the new labeling function, and the PN system remains non-codiagnosable.
- If t is uniquely observable for site 2, then the elementary F-path may not be eliminated since the triples are replaced by $(\varepsilon, t, \varepsilon)(t, \varepsilon, t)$ or $(t, \varepsilon, t)(\varepsilon, t, \varepsilon)$ under the new labeling function, and the PN system remains non-codiagnosable.
- If t is observable for both sites, then the elementary F-path may not be eliminated since the triples are replaced by (t, t, t) under the new labeling function, and the PN system remains non-codiagnosable.

Hence, condition C5 is necessary.

B. Results

We first prove that for site j, if rules R1 and R2 are applied incrementally, distinguishable sequences are still distinguishable.

Proposition 1: Let $\mathcal{L}'_1,\ldots, \mathcal{L}'_\nu$ be the labeling functions obtained from $\mathcal{L}_1,\ldots, \mathcal{L}_\nu$ by applying rules R1 and R2. For any site $j \in \mathcal{J}$, given two transition sequences $\sigma_1, \sigma_2 \in T^*$, it holds that $\mathcal{L}_j(\sigma_1) \neq \mathcal{L}_j(\sigma_2) \Rightarrow \mathcal{L}'_j(\sigma_1) \neq \mathcal{L}'_j(\sigma_2)$.

Proof. We prove this by contraposition, namely we prove that $\mathcal{L}'_j(\sigma_1) = \mathcal{L}'_j(\sigma_2) \Rightarrow \mathcal{L}_j(\sigma_1) = \mathcal{L}_j(\sigma_2)$.

We preliminarily notice that any unobservable transition sequence under $\mathcal{L}'_j$ is unobservable under $\mathcal{L}_j$. Similarly, two transitions that map to the same label in A_j under $\mathcal{L}'_j$ also map to the same label under $\mathcal{L}_j$.

The only changes in the two sequences σ_1 and σ_2 come from the transitions relabeled by $\mathcal{L}'_j$. Let t be one of such transitions. Two different cases may occur: i) If $t \in T_{r,o}$, it holds that $\mathcal{L}_j(t) = \varepsilon$; ii) If $t \in T_{j,o}$, it holds that $\mathcal{L}_j(t) \in A_j$. For both cases, two sequences σ_1 and σ_2 have the same observation under $\mathcal{L}_j$, i.e., $\mathcal{L}_j(\sigma_1) = \mathcal{L}_j(\sigma_2)$. Hence, the result holds. □

Proposition 2: There must exist at least one transition in each elementary F-path that can be relabeled in accordance with LO1 to LO7, rules R1, R2, and conditions C1 to C5.

Proof. By contradiction, we suppose that there is an elementary F-path that cannot be relabeled according to any rule. It means that in the path all transitions are in the form of (t, t, t) or (t, t, ε) or (t, ε, t) or LO7 or consecutive triples of the form $(\varepsilon, t, \varepsilon)(t, \varepsilon, \varepsilon)(\varepsilon, \varepsilon, t)$ or $(\varepsilon, \varepsilon, t)(t, \varepsilon, \varepsilon)(\varepsilon, t, \varepsilon)$. The case of LO7 can be omitted since a fault can not be relabeled. For the other cases, the elementary F-path may still exist even if all observable transitions are relabeled and each unobservable transition in $T_{r,o}$ is observable for both sites. Therefore, the PN is still non-codiagnosable under the labeling function $\mathcal{L}_{total}$ (defined by Eq. (5)), which contradicts Assumption A5. □

The following two propositions prove that elementary F-paths can be eliminated implementing options LO1 to LO7, rules R1 and R2, and guaranteeing the satisfaction of conditions C1 to C5. In addition, no new elementary F path is created.

Proposition 3: Let $\tau = (\gamma_{i_1}, \gamma_{j_1}, \gamma_{k_1})(\gamma_{i_2}, \gamma_{j_2}, \gamma_{k_2}) \dots (\gamma_{i_l}, \gamma_{j_l}, \gamma_{k_l})$ be an elementary F-path relabeled in accordance with LO1 to LO7, rules R1, R2, and conditions C1 to C5. Let $\sigma_i = \gamma_{i_1}\gamma_{i_2} \dots \gamma_{i_l}$, $\sigma_j = \gamma_{j_1}\gamma_{j_2} \dots \gamma_{j_l}$ and $\sigma_k = \gamma_{k_1}\gamma_{k_2} \dots \gamma_{k_l}$. It holds that: $\mathcal{L}_{1,new}(\sigma_i) \neq \mathcal{L}_{1,new}(\sigma_j)$ or $\mathcal{L}_{2,new}(\sigma_i) \neq \mathcal{L}_{2,new}(\sigma_k)$.

Proof. Since it is possible to relabel at least one transition in τ, we need to consider six different cases, i.e., LO1 to LO6.

Consider the case LO1 and let (t_i, t_j, t_k) be the triple of transitions of interest, where $t_i \in \sigma_i$, $t_j \in \sigma_j$, $t_k \in \sigma_k$ and $i = j = k$ does not hold. Since we assign to t_i a new label $\mathcal{L}_{1,new}(t_i) = \mathcal{L}_{2,new}(t_i) = t_i$, or/and to t_j a new label $\mathcal{L}_{1,new}(t_j) = \mathcal{L}_{2,new}(t_j) = t_j$, or/and to t_k a new label $\mathcal{L}_{1,new}(t_k) = \mathcal{L}_{2,new}(t_k) = t_k$, it must hold: $\mathcal{L}_{1,new}(\sigma_i) \neq \mathcal{L}_{1,new}(\sigma_j)$ or $\mathcal{L}_{2,new}(\sigma_i) \neq \mathcal{L}_{2,new}(\sigma_k)$.

Let us consider the case LO2 and let (t_i, t_j, ε) be the triple of transitions of interest, where $t_i \in \sigma_i$, $t_j \in \sigma_j$ and $i \neq j$. Since we assign to t_i a new label $\mathcal{L}_{1,new}(t_i) = t_i$, or/and to t_j a new label $\mathcal{L}_{1,new}(t_j) = t_j$, it must hold: $\mathcal{L}_{1,new}(\sigma_i) \neq \mathcal{L}_{1,new}(\sigma_j)$ or $\mathcal{L}_{2,new}(\sigma_i) \neq \mathcal{L}_{2,new}(\sigma_k)$. An analogous reasoning may be repeated for the case LO3.

Let us consider the case LO4 and let $(t, \varepsilon, \varepsilon)$ be the triple of transitions of interest, where $t \in \sigma_i$. Since we make t observable for site 1 or/and site 2 and we consider conditions C1 to C5, it cannot be synchronized with the transition t of nonfailure EBRGs wrt site 1 or/and site 2. Therefore, the result holds. Similar arguments can be repeated for LO5 and LO6. □

Proposition 4: If all elementary F-paths are relabeled in accordance with options LO1 to LO7, rules R1, R2 and conditions C1 to C5, then no elementary F-path wrt $\mathcal{L}_{new}$ is created.

Proof. By contradiction, we suppose that a new elementary F-path is created:

$\tau = (\gamma_{i_1}, \gamma_{j_1}, \gamma_{k_1})(\gamma_{i_2}, \gamma_{j_2}, \gamma_{k_2}) \dots (\gamma_{i_l}, \gamma_{j_l}, \gamma_{k_l})$

of length l wrt $\mathcal{L}_{new}$. Obviously, there exist three transition sequences $s_i, s_j, s_k \in L(N, m_0)$ satisfying:

i) $s_i \cap T_f \neq \emptyset$ and s_i is arbitrarily long after the fault.

ii) $s_j \cap T_f = \emptyset$, $s_k \cap T_f = \emptyset$.

iii) $\mathcal{L}_{1,new}(s_i) = \mathcal{L}_{1,new}(s_j)$ and $\mathcal{L}_{2,new}(s_i) = \mathcal{L}_{2,new}(s_k)$.

However, by Proposition 1, condition (iii) implies that $\mathcal{L}_1(s_i) = \mathcal{L}_1(s_j)$ and $\mathcal{L}_2(s_i) = \mathcal{L}_2(s_k)$. Therefore, the three sequences s_i, s_j, s_k form an elementary F-path under the initial labeling function $\mathcal{L}$. Since at least one transition in it has been relabeled in accordance with LO1 to LO6, rules R1, R2 and conditions C1 to C5, it holds that $\mathcal{L}_{1,new}(s_i) \neq \mathcal{L}_{1,new}(s_j)$ or $\mathcal{L}_{2,new}(s_i) \neq \mathcal{L}_{2,new}(s_k)$. Thus leading to a contradiction. □

Theorem 3: Let $(N, m_0, \mathcal{L})$ be a non-codiagnosable PN system satisfying assumptions A1 to A5. Let $\mathcal{L}_{new}$, $\mathcal{L}_{1,new}$ and $\mathcal{L}_{2,new}$ be the labeling functions obtained in accordance with rules R1, R2, relabeling options LO1 to LO7, and conditions C1 to C5. Then $(N, m_0, \mathcal{L}_{new})$ is codiagnosable.

Proof. This result is straightforward from Propositions 3 and 4. By Proposition 3, the relabeling procedure disables all elementary F-paths. By Proposition 4, no new path is created. Therefore, no elementary F-paths exist if they are all appropriately relabeled. By Theorem 2, the PN system $(N, m_0, \mathcal{L}_{new})$ is codiagnosable. □

VII. Optimal Relabeling Using Integer Nonlinear Programming

In this section, we show how to compute an optimal labeling function that minimizes the cost of the new sensors.

A. *Construction of nonlinear inequalities*

- Given $t \in T_o$, we define a binary variable $v_t \in \{0, 1\}$ as follows:

$$v_t = \begin{cases} 1, & \text{if } \mathcal{L}_{new}(t) = t \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

  In other words, $v_t = 1$ denotes that t is relabeled by $\mathcal{L}_{new}$.

- Similarly, given $t \in T_{r,o}$ and site $j \in \mathcal{J}$, we define a binary variable $v_t^j \in \{0, 1\}$ as follows (Note that the superscript j in v_t^j denotes site j):

$$v_t^j = \begin{cases} 1, & \text{if } \mathcal{L}_{j,new}(t) = t \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

  In other words, $v_t^j = 1$ means that t is observable for site j under $\mathcal{L}_{j,new}$. Moreover, according to Eq. (8), all the possible relabeling actions on $t \in T_{r,o}$ and their corresponding equations are detailed in Table III.

TABLE III: The relabeling actions on $t \in T_{r,o}$ and their corresponding equations.

Relabeling Action	Equation
t is uniquely relabeled by $\mathcal{L}_{1,new}$	$v_t^1 - v_t^1 v_t^2 = 1$
t is uniquely relabeled by $\mathcal{L}_{2,new}$	$v_t^2 - v_t^1 v_t^2 = 1$
t is relabeled by $\mathcal{L}_{1,new}$ and $\mathcal{L}_{2,new}$	$v_t^1 v_t^2 = 1$
t is relabeled by $\mathcal{L}_{1,new}$ or $\mathcal{L}_{2,new}$	$v_t^1 + v_t^2 - 2v_t^1 v_t^2 = 1$
t is relabeled by $\mathcal{L}_{1,new}$ or/and $\mathcal{L}_{2,new}$	$v_t^1 + v_t^2 - v_t^1 v_t^2 = 1$

For each elementary F-path τ, the following nonlinear inequality is constructed:

$$R_\tau \geq 1 \tag{9}$$

where R_τ is obtained by Algorithm 3.

In Algorithm 3, Line 4 to Line 29 correspond to relabeling options LO1 to LO7.

We use the notation $\mathcal{R}$ to denote the set of all nonlinear inequalities of the form Eq. (9) obtained with Algorithm 3 while examining each elementary F-path. The following theorem can be proved.

Theorem 4: Let $(N, m_0, \mathcal{L})$ be a non-codiagnosable PN system. There exist new labeling functions $\mathcal{L}_{new}$, $\mathcal{L}_{1,new}$, $\mathcal{L}_{2,new}$ that are computed eliminating all the elementary F-paths in the UV in accordance with rules R1, R2, relabeling options LO1 to LO7, and conditions C1 to C5, such that $(N, m_0, \mathcal{L}_{new})$ is codiagnosable iff $\mathcal{R}$ has a solution.

Proof. (Only if) By Algorithm 3, each inequality in $\mathcal{R}$ is obtained by implementing options LO1 to LO7, rules R1, R2 and conditions C1 to C5 to each elementary F-path. Given an elementary F-path, a transition $t \in T_{r,o}$ may be relabeled in accordance with rule R1 with LO4/LO5/LO6, while a transition $t \in T_o$ may be relabeled in accordance with rule R2 with LO1/LO2/LO3. In more detail,

- Rule R1 with LO4/LO5/LO6 are performed respectively by appending "$+v_t^1 + v_t^2 - v_t^1 v_t^2$" (Line 18), or "$+v_t^1$" (Line 23), or "$+v_t^2$" (Line 28) to the left of the inequality.
- Rule R2 with LO1/LO2/LO3 are performed respectively by appending "$+v_{t_i} + v_{t_j} + v_{t_k}$" (Line 6), or "$+v_{t_i} + v_{t_j}$" (Line 9), or "$+v_{t_i} + v_{t_k}$" (Line 12) to the left of the inequality.

Analogously,

- Condition C1 is performed by Line 22 and Line 16,
- Condition C2 is performed by Line 27 and Line 17,
- Condition C3 is performed by Line 26 and Line 14,
- Condition C4 is performed by Line 21 and Line 15,
- Condition C5 is performed by Line 25 and Line 20,

Algorithm 3: Construction of R_τ

```
Input: An elementary F-path $\tau$.
Output: $R_\tau$.
Let $R_\tau = 0$.
Examine each triple $(\gamma_i, \gamma_j, \gamma_k)$ in $\tau$ from its root to its leaf.
for each triple $(\gamma_i, \gamma_j, \gamma_k)$, do
    if LO1: $(\gamma_i, \gamma_j, \gamma_k) = (t_i, t_j, t_k)$, where $t_i, t_j, t_k \in T_o$, then
        if $i = j = k$, then do nothing,
        ;
        else $R_\tau = R_\tau + v_{t_i} + v_{t_j} + v_{t_k}$.;
    end
    if LO2: $(\gamma_i, \gamma_j, \gamma_k) = (t_i, t_j, \varepsilon)$, where $t_i, t_j \in T_o$, then
        if $i = j$, then do nothing,
        ;
        else $R_\tau = R_\tau + v_{t_i} + v_{t_j}$.;
    end
    if LO3: $(\gamma_i, \gamma_j, \gamma_k) = (t_i, \varepsilon, t_k)$, where $t_i, t_k \in T_o$, then
        if $i = k$, then do nothing,
        ;
        else $R_\tau = R_\tau + v_{t_i} + v_{t_k}$.;
    end
    if LO4: $(\gamma_i, \gamma_j, \gamma_k) = (t, \varepsilon, \varepsilon)$, where $t \in T_{r,o}$, then
        if C3: then ;
        its previous two consecutive triples in the path are $(\varepsilon, \varepsilon, t)(\varepsilon, t, \varepsilon)$ and $v_t^2 + v_t^1$ was added while
          examining the two triples, $R_\tau = R_\tau - v_t^1 - v_t^1 v_t^2$,
        else if C4: its previous two consecutive triples in the path are $(\varepsilon, t, \varepsilon)(\varepsilon, \varepsilon, t)$ and $v_t^1 + v_t^2$ was
          added while examining the two triples, then $R_\tau = R_\tau - v_t^2 - v_t^1 v_t^2$,;
        else if C1: its previous triple in the path is $(\varepsilon, t, \varepsilon)$ and $+v_t^1$ was added while examining the triple,
          then $R_\tau = R_\tau - v_t^1 + v_t^2$,;
        else if C2: its previous triple in the path is $(\varepsilon, \varepsilon, t)$ and $+v_t^2$ was added while examining the triple,
          then $R_\tau = R_\tau - v_t^2 + v_t^1$,;
        else $R_\tau = R_\tau + v_t^1 + v_t^2 - v_t^1 v_t^2$.;
    end
    if LO5: $(\gamma_i, \gamma_j, \gamma_k) = (\varepsilon, t, \varepsilon)$, where $t \in T_{r,o}$, then
        if C5: its previous two consecutive triples in the path are $(\varepsilon, \varepsilon, t)(t, \varepsilon, \varepsilon)$ and $+v_t^2 - v_t^2 + v_t^1$ was
          added while examining the two triples, then $R_\tau = R_\tau - v_t^1$,;
        else if C4: then ;
        its previous two consecutive triples in the path are $(t, \varepsilon, \varepsilon)(\varepsilon, \varepsilon, t)$ and $+v_t^1 + v_t^2 - v_t^1 v_t^2 - v_t^2 + v_t^1 v_t^2$
          was added while examining the two triples, $R_\tau = R_\tau - v_t^1 v_t^2$,
        else if C1: its previous triple in the path is $(t, \varepsilon, \varepsilon)$, $+v_t^1 + v_t^2 - v_t^1 v_t^2$ was added while examining
          the triple, then $R_\tau = R_\tau - v_t^1 + v_t^1 v_t^2$,;
        else $R_\tau = R_\tau + v_t^1$.;
    end
    if LO6:$(\gamma_i, \gamma_j, \gamma_k) = (\varepsilon, \varepsilon, t)$, where $t \in T_{r,o}$, then
        if C5: its previous two consecutive triple in the path are $(\varepsilon, t, \varepsilon)(t, \varepsilon, \varepsilon)$ and $+v_t^1 - v_t^1 + v_t^2$ was
          added while examining the two triples, then $R_\tau = R_\tau - v_t^2$,;
        else if C3: its previous two consecutive triples in the path are $(t, \varepsilon, \varepsilon)(\varepsilon, t, \varepsilon)$ and
          $+v_t^1 + v_t^2 - v_t^1 v_t^2 - v_t^1 + v_t^1 v_t^2$ was added while examining the two triples, then $R_\tau = R_\tau - v_t^1 v_t^2$,;
        else if C2: its previous triple in the path is $(t, \varepsilon, \varepsilon)$, $+v_t^1 + v_t^2 - v_t^1 v_t^2$ was added while examining
          the triple, then $R_\tau = R_\tau - v_t^2 + v_t^1 v_t^2$,;
        else $R_\tau = R_\tau + v_t^2$.;
    end
    if LO7: $\gamma_i \in T_f$, then do nothing.;
end
```

Finally, LO7 corresponds to Line 29, and "at least one transition should be relabeled" is performed by "≥ 1" in the inequality.

Hence, $\mathcal{R}$ has a solution if there exist new labeling functions $\mathcal{L}_{new}$, $\mathcal{L}_{1,new}$, $\mathcal{L}_{2,new}$ that are obtained by rules R1, R2, relabeling options LO1 to LO7, and conditions C1 to C5, with LO1 to LO7, such that $(N, m_0, \mathcal{L}_{new})$ is codiagnosable.

(If) Trivially derives from Theorem 3. □

B. Computation of optimal labeling functions by solving INLP

Typically, there exist more than one solution satisfying the set $\mathcal{R}$ of nonlinear inequalities. Here we introduce a performance index and compute a solution that minimizes it.

- Given $t \in T_{r,o}$, we denote by $(c_t^1\ c_t^2\ \ldots\ c_t^\nu)$ the *cost vector* of t, where c_t^j denotes the *cost* of associating a sensor to t, therefore its occurrence can be captured by site j.
- Given $t \in T_o$, c_t denotes the *cost* of associating a sensor to t, therefore it corresponds to a unique label that can be observed by the sites that are able to originally observe t.

In plain words, when a sensor is attached to an unobservable transition $t \in T_{r,o}$, the cost depends on the number of sites that are able to observe t under the new sensor configuration. On the other hand, when a new sensor is associated with a transition $t \in T_o$ that is already observable under $\mathcal{L}$, in order to make it distinguishable with respect to the other observable transitions, such a cost is only a function of t. Indeed, by Rule R2, we are assuming that the set of sensors that may observe a certain observable transition t does not change if t is relabeled.

By solving the following integer nonlinear programming (INLP), the new labeling functions $\mathcal{L}_{new}$, $\mathcal{L}_{1,new}$, $\mathcal{L}_{2,new}$ are obtained (The superscripts 1 and 2 denotes site 1 and 2, respectively):

$$\begin{cases} \min \displaystyle\sum_{t \in T_{r,o}} (c_t^1 v_t^1 + c_t^2 v_t^2) + \sum_{t \in T_o} c_t v_t \\ \text{s.t. } \ \mathcal{R}. \end{cases} \tag{10}$$

Example 4: Reconsider the PN in Fig. 1. For the sake of simplicity and without loss of generality, each transition in $T_{r,o} \cup T_o$ is assigned a unitary cost. For each elementary F-path in Fig. 4, we write an inequality according to Algorithm 3, thus $\mathcal{R}$ consists of the following nonlinear inequalities:

$v_{t_1}^1 + v_{t_1}^2 - v_{t_1}^1 v_{t_1}^2 - v_{t_1}^1 + v_{t_1}^1 v_{t_1}^2 - v_{t_1}^1 v_{t_1}^2 + v_{t_4}^2 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^1 + v_{t_1}^2 - v_{t_1}^1 v_{t_1}^2 - v_{t_1}^2 + v_{t_1}^1 v_{t_1}^2 - v_{t_1}^1 v_{t_1}^2 + v_{t_4}^2 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^1 + v_{t_1}^2 - v_{t_1}^1 v_{t_1}^2 - v_{t_1}^2 + v_{t_1}^1 v_{t_1}^2 + v_{t_4}^2 + v_{t_1}^1 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^1 - v_{t_1}^1 + v_{t_1}^2 - v_{t_1}^2 + v_{t_4}^2 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^1 + v_{t_1}^2 - v_{t_1}^2 - v_{t_1}^1 v_{t_1}^2 + v_{t_4}^2 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^1 + v_{t_1}^2 + v_{t_4}^2 + v_{t_1}^1 + v_{t_1}^2 - v_{t_1}^1 v_{t_1}^2 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^2 - v_{t_1}^2 + v_{t_1}^1 - v_{t_1}^1 + v_{t_4}^2 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^2 - v_{t_1}^2 + v_{t_1}^1 + v_{t_4}^2 + v_{t_1}^1 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^2 + v_{t_1}^1 - v_{t_1}^1 - v_{t_1}^1 v_{t_1}^2 + v_{t_4}^2 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^2 + v_{t_1}^1 + v_{t_4}^2 + v_{t_1}^1 + v_{t_1}^2 - v_{t_1}^1 v_{t_1}^2 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^2 + v_{t_4}^2 + v_{t_1}^1 + v_{t_1}^2 - v_{t_1}^1 v_{t_1}^2 - v_{t_1}^1 + v_{t_1}^1 v_{t_1}^2 + v_{t_8} + v_{t_{11}} \geq 1$
$v_{t_1}^2 + v_{t_4}^2 + v_{t_1}^1 - v_{t_1}^1 + v_{t_1}^2 + v_{t_8} + v_{t_{11}} \geq 1$

We solve the INLP problem (10) with the set of constraints $\mathcal{R}$ using the tool LINGO. We obtain the optimal solution: $\mathcal{L}_{1,new}(t_4) = \varepsilon$, $\mathcal{L}_{2,new}(t_4) = \mathcal{L}_{new}(t_4) = t_4$. Note that there exist other optimal solutions such as: $\mathcal{L}_{1,new}(t_8) = \mathcal{L}_{2,new}(t_8) = \mathcal{L}_{new}(t_8) = t_8$ or $\mathcal{L}_{1,new}(t_{11}) = \mathcal{L}_{2,new}(t_{11}) = \mathcal{L}_{new}(t_{11}) = t_{11}$.

Let us now consider the case of r fault classes. For each fault class T_f^i, we first construct an UV U_i by considering all faults in $T_f \setminus T_f^i$ as transitions to whom sensors cannot be attached. By looking at U_i, we compute the set of all elementary F-paths and construct the set of nonlinear inequalities $\mathcal{R}_i$ in accordance with Algorithm 3. The new labeling functions are finally obtained by solving INLP problem (10) under constraints $\mathcal{R}_i$ for $i = 1, 2, ..., r$.

In the case of ν local sites, we first construct the UV as the parallel composition of the EBRG and the nonfailure EBRGs wrt all the sites. Then, a set of nonlinear inequalities are obtained by applying relabeling rules to all the elementary F-paths in the UV. Note that $2\nu+3$ relabeling options are considered and rules R1, R2 are still available.

Besides, conditions are responsible for the elimination of each elementary F-path when some *consecutive arcs* exist (similar to conditions C1 to C5 in Section VI-A). Finally, the solution is obtained solving an INLP problem.

At the end of this section, we briefly discuss the complexity of the method. Generally speaking, the state space of the EBRG is much smaller than that of the reachability graph. By Algorithm 3, an elementary F-path contains at most $2x^{\nu+1}+1$ nodes, and a node has at most $(|T|+1)^{\nu+1}-1$ output arcs, where x is the number of states in G_e. Thus an UV contains at most $((|T|+1)^{\nu+1}-1)^{2x^{\nu+1}}$ elementary F-paths. For each elementary F-path, we need to build a nonlinear inequality according to Algorithm 3. Therefore, the complexity of generating an INLP is $O(((|T|+1)^{\nu+1}-1)^{2x^{\nu+1}})$.

VIII. Conclusions

The main contribution of this paper consists in proposing an approach to enforce codiagnosability to labeled Petri nets appropriately by adding sensors to transitions. We use the notion of Verifier and we unfold the Verifier to identify all paths violating the conditions for codiagnosability, and build an integer nonlinear programming problem to determine the set of optimal labeling functions. One of our future work consists in considering the optimal sensor selection problem under more general relabeling rules.

References

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

[2] 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, 2012.

[3] F. Basile, P. Chiacchio, and G. De Tommasi. On K-diagnosability of Petri nets via integer linear programming. *Automatica*, 48(9):2047 – 2058, 2012.

[4] R. Debouk, S. Lafortune, and D. Teneketzis. Coordinated Decentralized Protocols for Failure Diagnosis of Discrete Event Systems. *Discrete Event Dynamic Systems*, 10(1):33–86, 2000.

[5] M.V. Moreira, T.C. Jesus, and J.C. Basilio. Polynomial Time Verification of Decentralized Diagnosability of Discrete Event Systems. *IEEE Transactions on Automatic Control*, 56(7):1679–1684, 2011.

[6] M.P. Cabasino, A. Giua, A. Paoli, and C. Seatzu. Decentralized Diagnosis of Discrete-Event Systems Using Labeled Petri Nets. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(6):1477–1485, 2013.

[7] C. Keroglou and C.N. Hadjicostis. Distributed diagnosis using predetermined synchronization strategies. In *53rd IEEE Conference on Decision and Control*, pages 5955–5960, 2014.

[8] N. Ran, J. Hao, Z. Dong, Z. He, Z. Liu, Y. Ruan, and S. Wang. K-Codiagnosability Verification of Labeled Petri Nets. *IEEE Access*, 7:185055–185062, 2019.

[9] F. G. Cabral and M. V. Moreira. Synchronous Diagnosis of Discrete-Event Systems. *IEEE Transactions on Automation Science and Engineering*, 17(2):921–932, 2020.

[10] G.S. Viana, M.V. Moreira, and J.C. Basilio. Codiagnosability analysis of discrete-event systems modeled by weighted automata. *IEEE Transactions on Automatic Control*, 64(10):4361–4368, 2019.

[11] G.S. Viana and J.C. Basilio. Codiagnosability of discrete event systems revisited: A new necessary and sufficient condition and its applications. *Automatica*, 101:354 – 364, 2019.

[12] M.P. Cabasino, S. Lafortune, and C. Seatzu. Optimal sensor selection for ensuring diagnosability in labeled Petri nets. *Automatica*, 49(8):2373 – 2383, 2013.

[13] A. Giua, C. Seatzu, and D. Corona. Marking Estimation of Petri Nets With Silent Transitions. *IEEE Transactions on Automatic Control*, 52(9):1695–1699, 2007.

[14] M.P. Cabasino, A. Giua, and C. Seatzu. Fault detection for discrete event systems using Petri nets with unobservable transitions. *Automatica*, 46(9):1531 – 1539, 2010.

[15] M.P. Cabasino, A. Giua, M. Pocci, and C. Seatzu. Discrete event diagnosis using labeled Petri nets. An application to manufacturing systems. *Control Engineering Practice*, 19(9):989 – 1001, 2011.

[16] N. Ran, H. Su, A. Giua, and C. Seatzu. Codiagnosability Analysis of Bounded Petri Nets. *IEEE Transactions on Automatic Control*, 63(4):1192–1199, 2018.

[17] N. Ran, A. Giua, and C. Seatzu. Enforcement of Diagnosability in Labeled Petri Nets via Optimal Sensor Selection. *IEEE Transactions on Automatic Control*, 64(7):2997–3004, 2019.

[18] Z. Ma, X. Yin, and Z. Li. Marking diagnosability verification in labeled Petri nets. *Automatica*, 131:109713, 2021.

[19] N. Ran, J. Hao, and C. Seatzu. Prognosability analysis and enforcement of bounded labeled Petri nets. *IEEE Transactions on Automatic Control*, 2021.

[20] Z. Ma, Y. Tong, Z. Li, and A. Giua. Basis Marking Representation of Petri Net Reachability Spaces and Its Application to the Reachability Problem. *IEEE Transactions on Automatic Control*, 62(3):1078–1093, 2017.

[21] H. Lan, Y. Tong, J. Guo, and C. Seatzu. Verification of C-detectability using Petri nets. *Information Sciences*, 528:294–310, 2020.

[22] H. Lan, Y. Tong, and C. Seatzu. Analysis of strong and strong periodic detectability of bounded labeled petri nets. *Nonlinear Analysis: Hybrid Systems*, 42:101087, 2021.

[23] A. Madalinski and V. Khomenko. Predictability Verification with Parallel LTL-X Model Checking Based on Petri Net Unfoldings. *IFAC Proceedings Volumes*, 45(20):1232–1237, 2012.