Decentralized Diagnosability Analysis of Discrete Event Systems using Petri Nets

Maria Paola Cabasino, Alessandro Giua, Andrea Paoli, Carla Seatzu

Abstract

In this paper we present a procedure to analyze the diagnosability of a Petri net system in a decentralized framework. We recall the definition of failure ambiguous strings, i.e., strings that can be both faulty or not in the decentralized case, while can be distinguished in a centralized framework. We first prove that the absence of such kind of strings guarantees that the system is diagnosable in a decentralized framework. Then, we give an efficient procedure to verify the absence of such kind of strings for both bounded and unbounded Petri net systems.

Published as:

M.P. Cabasino, A. Giua, A. Paoli, C. Seatzu "Decentralized Diagnosability Analysis of Discrete Event Systems using Petri Nets," *13th IFAC World Congress*, August-September 2011.

This work has been partially supported by the European CommunityŠs Seventh Framework Programme under project DISC (Grant Agreement n. INFSO-ICT-224498).

M.P. Cabasino, A. Giua and C. Seatzu are with the Dept. of Electrical and Electronic Engineering, University of Cagliari, Italy, e-mail: {cabasino, giua, seatzu}@diee.unica.it.

A. Paoli is with the Department of Electronic, Computer Science and Systems, University of Bologna, Italy, e-mail: andrea.paoli@unibo.it.

I. Introduction

Failure detection is a crucial problem in automation systems. It has received a lot of attention in the past few decades. Diagnosis approaches can solve two different types of problems: the problem of diagnosis and the problem of diagnosability. Solving a problem of diagnosis means that we associate to each observed string of events a diagnosis state, such as "normal" or "faulty" or "uncertain". On the other hand, diagnosability implies the ability to locate a fault after a finite number of observations for *any* sequence (any behavior) of the system. Several contributions have been presented in the discrete event systems framework, both for automata (1; 2; 3; 4; 5) and Petri nets (PNs) (6; 7; 8; 9; 10)).

Due to the intrinsic distributed nature of the real systems, a lot of distributed diagnosis techniques, that take advantage of the natural decompositions of a modular system, have been studied both dealing with automata (11; 3; 12; 13; 14; 15; 16) and PNs (17; 18; 19).

In (20; 21) we presented an approach for decentralized diagnosis using Petri nets. The decentralized architecture that we used is composed by a set of sites communicating their diagnosis information with a coordinator that is responsible of detecting the occurrence of failures in the system. In particular, we defined a series of protocols that differ for the amount of information exchanged between the local sites and the coordinator, and the rules adopted by the coordinator to compute the global diagnosis states.

In this paper we consider the same decentralized architecture and we first introduce the definition of failure ambiguous strings. Secondly, we show that the absence of such kind of sequences is a sufficient condition for the diagnosability of a given net system in a decentralized framework, regardless of the considered protocol. We also discuss that, the absence of failure ambiguous strings is also a necessary condition for codiagnosability, i.e., diagnosability in the case in which there is no communication between the sites and the coordinator. Finally, we give a procedure to detect the presence of failure ambiguous strings based on the construction of a particular net called *Modified Verifier Net* (MVN), that is an extension of a particular net system, called *Verifier Net* (VN), that we introduced in (22) to analyze the diagnosability of an unbounded net system in a centralized framework.

Several polynomial time algorithms have been presented in the literature to detect failure ambiguous traces in the case of automata (14; 15; 16). In particular, (16) propose an algorithm to verify the codiagnosability that has a computational complexity lower than all the other methods previously proposed in literature. The main advantage of our approach with respect to (wrt) the automata ones is that it can be applied to systems having an infinite state space. Moreover, it can be shown that in the case of bounded systems the computational complexity of our approach is comparable to that of the automata approaches (14; 15; 16).

II. Background on labeled Petri nets

A *Place/Transition net* (P/T net) is a structure $N = (P, T, Pre, Post)$, where P is the set of m places, T is the set of n transitions, $Pre : P \times T \rightarrow \mathbb{N}$ and $Post : P \times T \rightarrow \mathbb{N}$ are the pre and post incidence functions that specify the arcs. The function $C = Post - Pre$ is called incidence matrix.

A *marking* is a vector $M : P \rightarrow \mathbb{N}$ that assigns to each place a nonnegative integer number of tokens; the marking of a place p is denoted with $M(p)$. A *net system* $\langle N, M_0 \rangle$ is a net N with initial marking M_0.

A transition t is enabled at M iff $M \geq Pre(\cdot, t)$ and may fire yielding the marking $M' = M + C(\cdot, t)$. The notation $M[\sigma\rangle$ is used to denote that the sequence of transitions $\sigma = t_1 \ldots t_k$ is enabled at M; moreover we write $M[\sigma\rangle M'$ to denote the fact that the firing of σ from M yields to M'. Given a sequence $\sigma \in T^*$ we write $t \in \sigma$ to denote that a transition t is contained in σ.

The set of all sequences that are enabled at the initial marking M_0 is denoted with $L(N, M_0)$. Given a sequence $\sigma \in T^\star$, we call $\pi : T^\star \rightarrow \mathbb{N}^n$ the function that associates to σ a vector $y \in \mathbb{N}^n$, named *firing vector*, such that $y(t) = k$ if the transition t is contained k times in σ.

A marking M is said to be *reachable* in $\langle N, M_0 \rangle$ iff there exists a firing sequence σ such that $M_0[\sigma\rangle M$. The set of all markings reachable from M_0 defines the *reachability set* of $\langle N, M_0 \rangle$ and is denoted with $R(N, M_0)$. Finally we define $PR(N, M_0)$ the potentially reachable set, i.e., the set of all markings $M \in \mathbb{N}^m$ for which there exists a vector $y \in \mathbb{N}^n$ that satisfies the *state equation* $M = M_0 + C \cdot y$. It holds that $R(N, M_0) \subseteq PR(N, M_0)$.

A net system $\langle N, M_0 \rangle$ is said to be bounded if there exists a positive constant k such that for all $M \in R(N, M_0)$, $M(p) \leq k$. If such is not the case, namely if the number of tokens in one or more places can grow indefinitely, then the Petri net system is *unbounded*.

A sequence $\sigma \in T^*$ is called *repetitive* if there exists a marking $M_1 \in R(N, M_0)$ such that $M_1[\sigma\rangle M_2[\sigma\rangle M_3[\sigma\rangle \cdots$, i.e., if it can fire infinitely often starting from M_1.

A *labeling function* $\mathcal{L} : T \to L \cup \{\varepsilon\}$ assigns to each transition a symbol from a given alphabet L or the empty word ε. The set of transitions sharing the same label e is denoted as T_e. Transitions whose label is ε are called *silent* and are denoted by the set T_u. The set $T_o = T \setminus T_u$ is the set of *observable transitions*, i.e., when an observable transition fires we observe its label. We denote as C_u (C_o) the restriction of the incidence matrix to T_u (T_o).

Finally, given a net $N = (P, T, Pre, Post)$ and a subset $T' \subseteq T$ of its transitions, we define the T'-induced subnet of N as the new net $N' = (P, T', Pre', Post')$, where Pre' and $Post'$ are the restrictions of Pre and $Post$ to T', i.e., N' is the net obtained from N removing all transitions in $T \setminus T'$. We write that $N' \prec_{T'} N$.

III. THE DECENTRALIZED DIAGNOSIS ARCHITECTURE

We model anomalous or faulty behavior using the set of silent transitions $T_f \subseteq T_u$. The set T_f includes all fault transitions and is further decomposed into r different subsets T_f^i, where $i \in \mathcal{F} = \{1, \ldots, r\}$, that model different fault classes. As in most of the literature in this topic, we assume that the fault model is known. The transition set $T_{reg} = T_u \setminus T_f$ represents the set of unobservable, but regular, transitions.

The problem of fault diagnosis can be seen as the problem of detecting the firing of any fault transition in T_f, using the knowledge of the firing of observable transitions, or the knowledge of their labels in the case of labeled PNs.

In this work we analyze the decentralized diagnosability properties of a system using a decentralized architecture as depicted in Fig. 1. The system is monitored by a set $\mathcal{J} = \{1, \ldots, \nu\}$ of sites. Each site has a complete knowledge of the net structure and of the initial marking, but observes the evolution of the system using its own observation mask. Obviously, different sites have different observation masks. In particular, for any site $j \in \mathcal{J}$, the set of locally observable transitions is the set $T_{o,j} \subseteq T_o$. Any centrally observable transition is observed by at least one site, i.e., $\bigcup_{j \in \mathcal{J}} T_{o,j} = T_o$. The set of locally unobservable transitions is defined as

$$T_{u,j} = T_{reg} \cup T_f \cup (T_o \setminus T_{o,j}). \tag{1}$$

We denote as

$$\bar{\mathcal{L}} : T \to L \cup \{\varepsilon\} \tag{2}$$

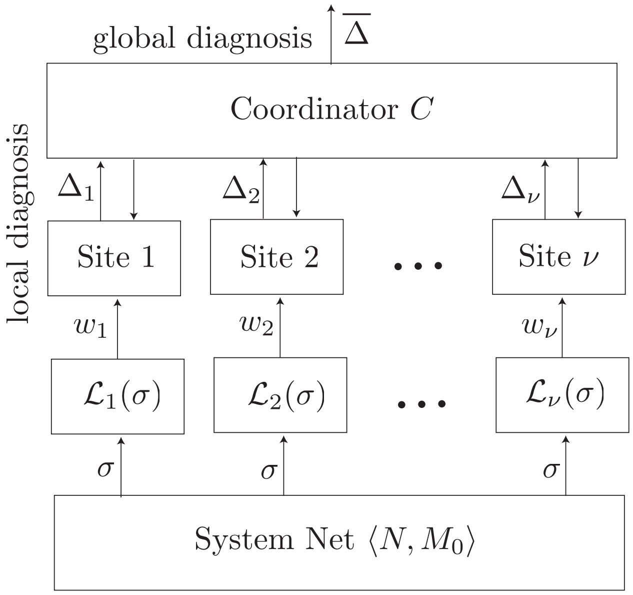

Fig. 1. The decentralized diagnosis architecture.

the labeling function associated to the centralized system, namely a system that is able to observe all labels associated to transitions in T_o. Moreover, for all $j \in \mathcal{J}$, $L_j \subseteq L$ denotes the alphabet of the j-th site, i.e., the set of labels observable by the j-th site, and

$$\mathcal{L}_j(t) = \begin{cases} \bar{\mathcal{L}}(t) & \text{if } \bar{\mathcal{L}}(t) \in L_j \\ \varepsilon & \text{otherwise} \end{cases} \tag{3}$$

is the labeling function associated to the j-th site. Note that the above definition of $\mathcal{L}_j$ implies that, if a site observes a transition labeled e, then it observes all transitions whose label is e. Finally, $w_j = \mathcal{L}_j(\sigma)$ denotes the word of events in L_j associated to the sequence σ by the j-th site.

In this paper we do not care which kind of protocol will be used for exchanging information between the coordinator and the local sites. We just suppose that each local site performs a local diagnosis, using a given diagnosis approach, and depending on this, it exchanges information with a *coordinator* C according to a given protocol. In the most elementary protocol the coordinator diagnoses a fault in a given fault class as soon as one site communicates the detection of it. More precisely, for each fault class $i \in \mathcal{F}$ it computes a diagnosis state $\bar{\Delta}_i$ that can be "normal" or "faulty" or "uncertain". We can also think that the sites communicate to the coordinator not only when a fault occurs, but also other additional information helping in the diagnosis. In the borderline case all sites communicate to the coordinator the observed sequences, and if the coordinator knows the net structure and the sites observation masks it acts as a centralized observer. In this paper where not specified we will refer to the most elementary protocol case.

IV. Diagnosability and failure ambiguous strings

In this section we first recall the notion of diagnosability, then we introduce the definition of failure ambiguous strings, and show the relationships among them. In particular, as it is common in all the literature in this topic, we make the following assumption.

A1 The system does not enter a deadlock after the firing of any fault transition.

Based on such an assumption, the following definition can be given.

Definition *4.1:* Let us consider a PN system $\langle N, M_0\rangle$ having no deadlock after the occurrence of transition $t_f \in T_f^i$, for all $i \in \mathcal{F}$. Assume that diagnosis is performed according to a given approach (either centralized or decentralized).

We say that $\langle N, M_0\rangle$ is *diagnosable wrt the fault class* T_f^i *and wrt a given diagnosis approach* iff the occurrence of some fault in T_f^i is unambiguously detected using the specified diagnosis approach after a *finite* number of transition firings. ■

Definition *4.2:* A PN system $\langle N, M_0\rangle$ is *diagnosable wrt a given diagnosis approach* if it is diagnosable wrt that approach for all fault classes T_f^i, $i \in \mathcal{F}$. ■

Note that in the centralized framework, inspired by the definition of diagnosability for languages introduced in (23), Definition 4.1 can alternatively be formulated as follows.

Definition *4.3:* A PN system $\langle N, M_0\rangle$ having no deadlock after the occurrence of transition $t_f \in T_f^i$, for $i \in \mathcal{F}$, is *diagnosable wrt the fault class* T_f^i if there do not exist two firing sequences σ_1 and $\sigma_2 \in T^*$ satisfying the following conditions:

- $\mathcal{L}(\sigma_1) = \mathcal{L}(\sigma_2)$,
- $\forall t_f \in T_f^i$, $\sigma_1 \in (T \setminus T_f^i)^*$,
- $\exists$ at least one $t_f \in T_f^i$ such that $t_f \in \sigma_2$,
- σ_2 is of "arbitrary length" after fault $t_f \in T_f^i$, i.e., there exists at least one decomposition $\sigma_2 = \sigma_2' t_f \sigma_2''$ such that given any $k \in \mathbb{N}$ you can always pick σ_2'' such that $|\sigma_2''| > k$. ■

Several protocols can be defined, based on the decentralized diagnosis architecture introduced in the previous section, that basically differ for the kind and the amount of information exchanged between the local sites and the coordinator. In this section we want to show that, in all such cases, when analyzing diagnosability in a decentralized framework, the first important step is that of detecting the presence of particular strings, called *failure ambiguous strings*.

Note that the notion of failure ambiguous strings has been firstly introduced in (11) in the framework of automata under the assumption of two sites. Here we extend such definition to PNs and consider the general case of an arbitrary number ν of sites.

Definition *4.4:* Consider a net system $\langle N, M_0 \rangle$ monitored by a set $\mathcal{J} = \{1, \ldots, \nu\}$ of sites. Let $T_{o,j} \subseteq T_o$ be the set of locally observable transitions for the generic site $j \in \mathcal{J}$. Finally, let $T_f^i \subseteq T_f$ be the generic i-th fault class, with $i \in \mathcal{F}$.

A string $\sigma \in T^*$ such that $t_f \in \sigma$ for at least one $t_f \in T_f^i$, is said to be *failure ambiguous* wrt the above set of sites and wrt the fault class T_f^i, if the following two conditions are verified:

(a) $\mathcal{L}_j^{-1}(\mathcal{L}_j(\sigma)) \cap (T \setminus T_f^i)^* \neq \emptyset \quad \forall j \in \mathcal{J}$;

(b) $\bar{\mathcal{L}}^{-1}(\bar{\mathcal{L}}(\sigma)) \cap (T \setminus T_f^i)^* = \emptyset$,

where $\mathcal{L}_j$ and $\bar{\mathcal{L}}$ are defined as in (3) and (2), respectively.

■

In simple words, a sequence σ containing some fault transitions in a fault class i, is failure ambiguous wrt to a set of sites and wrt the i-th fault class, if the word σ is ambiguous for each site $j \in \mathcal{J}$, i.e., it may also be explained by a non faulty word, while the word σ is not ambiguous for the centralized system.

Example *4.5:* Let us consider the PN system in Fig. 2 which is locally diagnosed by two sites whose alphabets are equal to $L_1 = \{a, c\}$ and $L_2 = \{b, c\}$, respectively. The sequence $\sigma = t_f t_1 t_2 t_3^q$, with $q \in \mathbb{N}$, is failure ambiguous wrt the sites 1 and 2 and wrt to the unique fault class $T_f = \{t_f\}$. In fact, $\mathcal{L}_1(\sigma) = \{ac^q\}$ and $\mathcal{L}_1^{-1}(\mathcal{L}_1(\sigma)) = \{t_f t_1 t_2 t_3^q, t_5 t_3^q\}$, thus $\mathcal{L}_1^{-1}(\mathcal{L}_1(\sigma)) \cap (T \setminus T_f)^* = \{t_5 t_3^q\}$; $\mathcal{L}_2(\sigma) = \{bc^q\}$ and $\mathcal{L}_2^{-1}(\mathcal{L}_2(\sigma)) = \{t_f t_1 t_2 t_3^q, t_4 t_3^q\}$ thus $\mathcal{L}_2^{-1}(\mathcal{L}_2(\sigma)) \cap (T \setminus T_f)^* = \{t_4 t_3^q\}$; and $\bar{\mathcal{L}}(\sigma) = \{abc^q\}$ and $\bar{\mathcal{L}}^{-1}(\bar{\mathcal{L}}(\sigma)) = \{t_f t_1 t_2 t_3^q\}$ thus $\bar{\mathcal{L}}^{-1}(\bar{\mathcal{L}}(\sigma)) \cap (T \setminus T_f^i)^* = \emptyset$.

■

Obviously, regardless of the considered protocol, if a system is diagnosable in a centralized framework with respect to a given fault class, and has no failure ambiguous string of arbitrary length with respect to that class, it is also diagnosable in a decentralized framework. The following proposition formally proves this.

Proposition *4.6:* Consider a net system $\langle N, M_0 \rangle$ monitored by a set $\mathcal{J} = \{1, \ldots, \nu\}$ of sites.

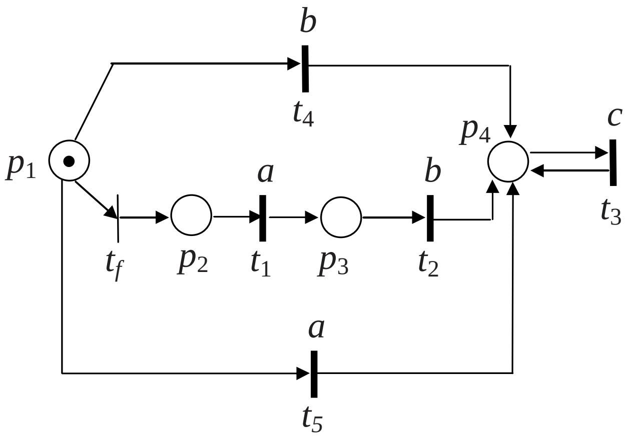

Fig. 2. The PN system considered in Example 4.5.

Let $T_f^i \subseteq T_f$ be the generic i-th fault class, with $i \in \mathcal{F}$. Let us suppose that the net system $\langle N, M_0 \rangle$ is diagnosable in a centralized framework wrt T_f^i.

Regardless of the protocol used to perform decentralized diagnosis, if there do not exist failure ambiguous strings of arbitrary length for the considered set of sites wrt to T_f^i, then the system is also diagnosable in a decentralized framework.

Proof: By Definition 4.4, if there do not exist failure ambiguous strings of arbitrary length wrt a given fault class, it means that there do not exist strings of arbitrary length that can be distinguished by the centralized diagnoser, but cannot be distinguished by all the local sites. This implies that, for each string containing a fault there exists at least one site that detects the fault. Thus the system is diagnosable also in a decentralized framework. □

Note that in general cases, as it happens in the case of automata (11), the absence of failure ambiguous strings of arbitrary length is only a sufficient condition for the diagnosability in a decentralized framework. In fact, if protocols are defined so that local sites take advantage of the information collected by the other sites (e.g., receiving certain information by the coordinator), the resulting system may be diagnosable even in the presence of failure ambiguous strings. On the contrary, if each site computes its diagnosis states receiving no information from the other sites and from the coordinator, then the absence of failure ambiguous strings is also a necessary condition for the decentralized diagnosability.

As an example, in (20; 21), we presented three different protocols for the decentralized diagnosis using labeled PNs. In the easiest one, denoted as Protocol 1, a site communicates to the coordinator its diagnosis state if and only if it has detected the occurrence of a fault. No communication is allowed among sites, and among the coordinator and the local sites. The coordinator simply produces an alarm if it receives it by at least one site. In such a case it is obvious that the absence of failure ambiguous strings is not only a sufficient condition for decentralized diagnosability, but it is also necessary. On the contrary, if we use the more sophisticated protocols, denoted as Protocol 2 and 3, it may occur that a system is diagnosable in a decentralized framework even in the presence of failure ambiguous strings. This is due to the fact that the protocol is based on a confutation procedure that allows the sites to take benefit of the information sent by the other sites to the coordinator.

V. Detecting failure ambiguous strings using the Modified Verifier Net

In this section we provide a procedure to determine if a given net system (bounded or unbounded) observed by ν local sites has failure ambiguous strings of arbitrary length. The proposed procedure is based on the definition of a particular PN system that we call *Modified Verifier Net* (MVN) since it is a generalization of a particular finite state automaton called *Verifier Net* (VN) that we used in (22) to analyze the diagnosability of unbounded PNs in a centralized framework.

A. Modified Verifier Net

For the sake of simplicity, we consider the case of a single fault class, hence the superscript i is omitted in T_f^i hereafter.

The *Modified Verifier Net* system is the PN system obtained by a composition (related to parallel composition) performed on the transitions labels of the centralized system $\langle N, M_0 \rangle$ with labeling function $\mathcal{L}$ and ν net systems $\langle N^j, M_0^j \rangle$, where $j = \{1, \ldots, \nu\}$, obtained by $\langle N, M_0 \rangle$ removing the fault and having a labeling function $\mathcal{L}^j$. Thus we define:

- $\langle N, M_0 \rangle$ as the centralized PN system where $N = (P, T, Pre, Post)$, $T = T_o \cup T_u$, and $T_u = T_{reg} \cup T_f$ and labeling function $\mathcal{L} \ : \ T \to L \cup \{\varepsilon\}$;
- $\langle N^j, M_0^j \rangle$ as the PN systems associated to the local sites ($j \in \mathcal{J}$) where $N^j = (P^j, T^j, Pre^j, Post^j)$ is the T^j-induced subnet, $P^j = P$, $T^j = T \setminus T_f$, Pre^j and $Post^j$ are the restrictions of Pre

and $Post$ to P^j and T^j and $M_0^j = M_0$ for all $j \in \mathcal{J}$, with labeling function $\mathcal{L}^j$ restricted to T^j.

Note that $\mathcal{L}^j$ and $\mathcal{L}_j$ denote two different labeling functions: $\mathcal{L}^j$ is the generic labeling function associated to the PN system $\langle N^j, M_0^j \rangle$ that maps each transition in $T \setminus T_f$ with its proper label, while $\mathcal{L}_j$ is defined as in Eq. (3). Therefore, $\mathcal{L}^j$ is the same for all the sites, while $\mathcal{L}_j$ depends on the particular site.

Let $T_{reg}^j = T^j \cap T_{u,j}$ be the set of unobservable transitions of the jth site[1]. We denote the MVN as $\langle \tilde{N}, \tilde{M}_0 \rangle$, where $\tilde{N} = (\tilde{P}, \tilde{T}, \tilde{Pre}, \tilde{Post})$, $\tilde{P} = P \cup P^1 \cup \ldots P^\nu$ and

$$\begin{aligned} \tilde{T} \;\; &= \;\; (\cup_{e \in L} \tilde{T}_{o,e}) \cup (T_{reg} \times \{\lambda\} \times \ldots \times \{\lambda\}) \cup \\ & \qquad (\cup_{j=1,\ldots,\nu} \tilde{T}_{reg}^j) \cup (T_f \times \{\lambda\} \times \ldots \times \{\lambda\}), \end{aligned}$$

where

$$\tilde{T}_{o,e} = \left\{ \begin{array}{lll} (t, \gamma_1, \ldots, \gamma_\nu) & | & t \in T_e, \\ & & \gamma_j = \left\{ \begin{array}{ll} t^j \in T_{o,j}^j & \text{if } \mathcal{L}_j(t^j) = \mathcal{L}(t) \\ \lambda & \text{otherwise} \end{array} \right. \end{array} \right\}$$

and

$$\begin{aligned} \tilde{T}_{reg}^j = \{(\{\lambda\}, \beta_1, \ldots, \beta_\nu) \quad | \quad & \beta_j \in T_{reg}^j, \\ & \beta_i = \lambda \;\forall i \neq j\} \,. \end{aligned}$$

Matrices $\tilde{Pre}$ and $\tilde{Post}$ are the Pre and $Post$ matrices of the MVN and they are defined together with the rules of construction of the MVN in the following algorithm.

Algorithm 5.1: **Construction of the MVN.**

Input: a labeled PN system $\langle N, M_0 \rangle$ where $N = (P, T, Pre, Post)$, $T = T_o \cup T_{reg} \cup T_f$ and $\mathcal{L} : T \to L \cup \{\varepsilon\}$; a set of ν local sites S_j, $j = 1, \ldots, \nu$, with observable transitions $T_{o,j} \subsetneq T_o$.

Output: the MVN system $\langle \tilde{N}, \tilde{M}_0 \rangle$, where $\tilde{N} = (\tilde{P}, \tilde{T}, \tilde{Pre}, \tilde{Post})$.

1) Let $\langle N^j, M_0^j \rangle$, with $j = 1, \ldots, \nu$, be the labeled PN systems defined as discussed above.
2) Let $\tilde{P} = P \cup P^1 \cup \ldots \cup P^\nu$.
3) Let $\tilde{M}_0 = \begin{bmatrix} M_0 \\ M_0^1 \\ \vdots \\ M_0^\nu \end{bmatrix}$.

[1] Since T^j does not contain fault transitions, being by definition $T^j = T \setminus T_f$, it holds $T_{reg}^j \equiv T_{u,j}$.

4) For all transitions $t_f \in T_f$, do
    - add a transition $t \in \tilde{T}$ denoted as $(t_f, \lambda, \ldots, \lambda)$;
    - for all $j = 1, \ldots, \nu$, do
        - for all $p \in P^j$, let $\tilde{Pre}(p,t) = \tilde{Post}(p,t) = 0$;
    - for all $p \in P$, let $\tilde{Pre}(p,t) = Pre(p,t_f)$ and $\tilde{Post}(p,t) = Post(p,t_f)$.
5) For all transitions $t_{reg} \in T_{reg}$,
    - add a transition $t \in \tilde{T}$ denoted as $(t_{reg}, \lambda, \ldots, \lambda)$;
    - for all $j = 1, \ldots, \nu$, do
        - for all $p \in P^j$, let $\tilde{Pre}(p,t) = \tilde{Post}(p,t) = 0$;
    - for all $p \in P$, let $\tilde{Pre}(p,t) = Pre(p,t_{reg})$ and $\tilde{Post}(p,t) = Post(p,t_{reg})$.
6) For all $j = 1, \ldots, \nu$, do
    - for all transitions $t^j_{reg} \in T^j_{reg}$, do
        - add a transition $t \in \tilde{T}$ denoted as
          $(\lambda, \beta_1, \ldots, \beta_\nu)$ where $\beta_j = t^j_{reg}$ and $\beta_i = \lambda$ for all $i \neq j$;
        - for all $p \in P^j$, let $\tilde{Pre}(p,t) = Pre^j(p,t^j_{reg})$ and $\tilde{Post}(p,t) = Post^j(p,t^j_{reg})$;
        - for all $p \in P^i \cup P$, with $i \neq j$, let $\tilde{Pre}(p,t) = \tilde{Post}(p,t) = 0$;
7) For all labels $e \in L$, do
    - for all $j = 1, \ldots, \nu$, let $T^j_e = \{t^j \in T^j_{o,j} \mid \mathcal{L}_j(t^j) = e\}$;
    - for all possible combinations of transitions in T_e and in T^j_e, $j = 1, \ldots, \nu$, do
        - add a transition $t \in \tilde{T}$ denoted as
          $(t_e, \gamma_1, \ldots, \gamma_\nu)$ where $t_e \in T_e$, $\gamma_j \in T^j_e$ if $T^j_e \neq \emptyset$, else $\gamma_j = \lambda$;
        - for all $p \in P$, let $\tilde{Pre}(p,t_e) = Pre(p,t_e)$, $\tilde{Post}(p,t_e) = Post(p,t_e)$;
        - for all $j = 1, \ldots, \nu$, do
            - ◇ for all $p \in P^j$, if $\gamma_j = \lambda$ let $\tilde{Pre}(p,t) = \tilde{Post}(p,t) = 0$, else let $\tilde{Pre}(p,t) = Pre^j(p,\gamma_j)$, $\tilde{Post}(p,t) = Post^j(p,\gamma_j)$;
        - label transition t with $(e, \varrho_1, \ldots, \varrho_\nu)$ where $\varrho_j = e$ if $T^j_e \neq \emptyset$, else $\varrho_j = \lambda$.

■

The following example presents an application of the above algorithm.

***Example** 5.2:* Let us consider the labeled PN system in Fig. 2. Let $T_o = \{t_1, t_2, t_3, t_4, t_5\}$, $T_f = \{t_f\}$ and $T_{reg} = \emptyset$. Moreover, let $\bar{\mathcal{L}}(t_1) = \bar{\mathcal{L}}(t_5) = a$, $\bar{\mathcal{L}}(t_2) = \bar{\mathcal{L}}(t_4) = b$, $\bar{\mathcal{L}}(t_3) = c$.

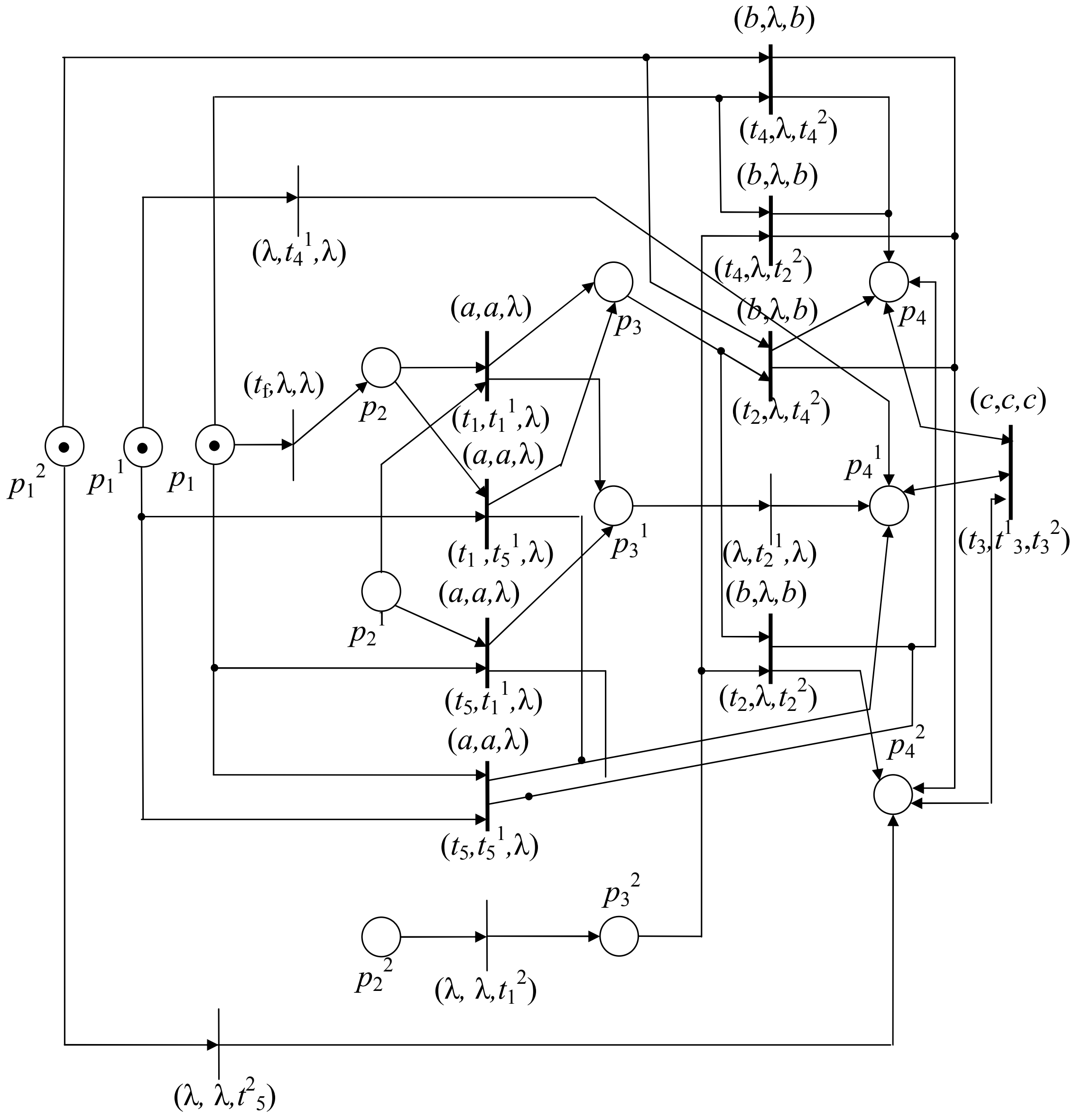

Fig. 3. The MVN built starting from the PN system in Fig. 2.

Assume that there are two local sites S_1 and S_2 whose set of observable events is $T_{o,1} = \{t_1, t_3, t_5\}$ and $T_{o,2} = \{t_2, t_3, t_4\}$, respectively. Thus $L_1 = \{a, c\}$ and $L_2 = \{b, c\}$.

As discussed above, all transitions that belong to T_o, but are not observable by a given site are considered as regular unobservable transitions by such a site.

The resulting MVN is reported in Fig. 3. The cardinality of $\tilde{P}$ is 12 since N has 4 places and the system is locally observed by two sites. The initial marking assigns one token to p_1, p_1^1, p_1^2

and no token to the other places.

Now, at Step 4 of Algorithm 5.1 only one transition is added to the MVN being $T_f = \{t_f\}$. In particular, the transition originating from t_f is (t_f, λ, λ).

Step 5, being $T_{reg} = \emptyset$, introduces no transition in the MVN.

On the contrary, Step 6 leads to the addition of four transitions. In particular, being $T^1_{reg} = \{t^1_2, t^1_4\}$ this leads to the addition of $(\lambda, t^1_2, \lambda)$ and $(\lambda, t^1_4, \lambda)$ in the MVN. These transitions are only connected to places in P^1 and their input and output arcs have the same weight of the input and output arcs of t_2 and t_4, respectively, in N. Analogously, being $T^2_{reg} = \{t^2_1, t^2_5\}$ at Step 6 we add two transitions in the MVN, namely $(\lambda, \lambda, t^2_1)$ and $(\lambda, \lambda, t^2_5)$. No arcs go from these transitions to places in P and P^1, while they are connected to places in P^2. In particular, their input and output arcs to places in P^2 have the same weight of the input and output arcs of t_1 and t_5, respectively, in N.

Finally, at Step 7 we add 9 transitions: four relative to label a, four relative to b and one to c. In particular, being $T_a = \{t_1, t_5\}$, $T^1_a = \{t^1_1, t^1_5\}$ and $T^2_a = \emptyset$, we consider all possible combinations of transitions in T_a and T^1_a and add a new transition for each of such combinations. Thus the following four transitions in the MVN correspond to label a: (t_1, t^1_1, λ), (t_1, t^1_5, λ), (t_5, t^1_1, λ) and (t_5, t^1_5, λ). All of them are labeled (a, a, λ). Analogously, being $T_b = \{t_2, t_4\}$, $T^1_b = \emptyset$ and $T^2_b = \{t^2_2, t^2_4\}$, in the MVN there are four transitions relative to b, namely, (t_2, λ, t^2_2), (t_2, λ, t^2_4), (t_4, λ, t^2_2) and (t_4, λ, t^2_4), all labeled (b, λ, b). Finally, being $T_c = \{t_3\}$, $T^1_c = \{t^1_3\}$ and $T^2_c = \{t^2_3\}$, the MVN has only one transition relative to c, i.e., (t_3, t^1_3, t^2_3) and such a transition is labeled (c, c, c). ■

***Proposition** 5.3:* Given a PN system $\langle N, M_0\rangle$, a set of ν local sites, and the corresponding MVN, if a sequence

$$\begin{aligned}\tilde{\sigma} \;&=\; (\gamma_{i_1}, \gamma^1_{i_1}, \ldots, \gamma^\nu_{i_1})(\gamma_{i_2}, \gamma^1_{i_2}, \ldots, \gamma^\nu_{i_2}) \cdots \\ &\qquad \cdots (\gamma_{i_k}, \gamma^1_{i_k}, \ldots, \gamma^\nu_{i_k}) \in \tilde{T}^*\end{aligned}$$

is repetitive in the MVN[2], then there exists a repetitive sequence $\sigma = \gamma_{i_1}\gamma_{i_2}\ldots\gamma_{i_k}$ in $\langle N, M_0\rangle$ and a repetitive sequence $\sigma^j = \gamma^j_{i_1}\ldots\gamma^j_{i_k}$ in $\langle N^j, M^j_0\rangle$, for all $j = 1, \ldots, \nu$.

Proof: It follows from the definition of MVN. In fact, the existence of a sequence $\tilde{\sigma} \in L(\tilde{N}, \tilde{M}_0)$ implies that $\sigma \in L(N, M_0)$ and $\sigma^j \in L(N^j, M^j_0)$, for all $j = 1, \ldots, \nu$. The firing

[2] The symbol λ denotes the sequence of length zero, hence $\sigma'\lambda\sigma'' = \sigma'\sigma''$.

sequences σ and σ^j, $j = 1, \ldots, \nu$, are repetitive respectively in $\langle N, M_0 \rangle$ and in $\langle N^j, M_0^j \rangle$, given that $\tilde{\sigma}$ is repetitive in the MVN. □

B. Detection of failure ambiguous strings

We now provide a constructive criterion to establish if a given net system presents failure ambiguous strings of arbitrary length by looking at the MVN. In particular, we make the following assumption.

A2 The net system $\langle N, M_0 \rangle$ with set of observable transitions T_o, labeling function $\bar{\mathcal{L}}$ and set of regular unobservable transitions T_{reg}, is diagnosable in a centralized framework.

Note that this is not a restrictive assumption because if the system is not diagnosable in a centralized framework there is no interest in determining the existence of failure ambiguous strings.

The criterion we propose is based on the following proposition.

***Proposition** 5.4:* Let $\langle N, M_0 \rangle$ be a PN system monitored by a set $\mathcal{J} = \{1, \ldots, \nu\}$ of local sites. Let $T_{o,j} \subsetneq T_o$ be the set of locally observable transitions for the generic site $j \in \mathcal{J}$ and $T_o = \cup_{j \in \mathcal{J}} T_{o,j}$. Let assumptions **A1** and **A2** be verified.

Assume for simplicity that there exists only one fault class T_f having only one fault transition, i.e., $T_f = \{t_f\}$.

There exist no failure ambiguous strings of arbitrary length wrt the above set of sites and wrt T_f iff starting from any node of the reachability/converability graph (R/CG) of its MVN reached by firing the fault there does not exist any cycle associated with a repetitive sequence in the MVN.

Proof: Let us preliminary observe that by assumption **A1** the system does not enter a deadlock after the firing of any fault transition, i.e., for any fault transition, there exists at least one sequence of arbitrary length containing it. Therefore, a deadlock in the MVN occurring after t_f, cannot be due to the fact that the deadlock really occurs in the net after the firing of t_f.

Let us now prove the *if* and *only if* statements separately.

(If) We prove this by contradiction. By Proposition 5.3, if the R/CG of the MVN has a cycle associated to a repetitive sequence after some occurrence of t_f, it means that there exists at least one sequence of arbitrary length $\sigma \in T^*$ containing t_f that is enabled by $\langle N, M_0 \rangle$, and other sequences $\sigma^j \in (T^j)^*$, one for each site $j \in \mathcal{J}$, having the same observable projection of σ, i.e.,

$\mathcal{L}_j(\sigma) = \mathcal{L}_j(\sigma^j)$, that are enabled by $\langle N^j, M_0^j \rangle$. However, by assumption **A2**, this implies that σ is a failure ambiguous string of arbitrary length wrt the considered set of local sites and the fault transition t_f, thus leading to a contradiction.

(Only if) We now show that if there do not exist failure ambiguous strings of arbitrary length wrt the set of sites $\mathcal{J} = \{1, \ldots, \nu\}$ and wrt T_f, then the R/CG of the MVN has no cycle associated to a repetitive sequence after some occurrence of t_f.

If there do not exist failure ambiguous strings wrt the local sites and wrt T_f, it means that we may have a string σ containing t_f and other ν strings σ^j that do not contain t_f and such that $\mathcal{L}_j(\sigma) = \mathcal{L}_j(\sigma^j)$, but we will never be able to extend them arbitrarily while keeping their projections identical. However, by construction of the MVN, this means that the R/CG of the MVN will have no cycle associated to a repetitive sequence after any occurrence of t_f, thus proving the statement. □

Obviously, in the case of more than one fault class, a different MVN should be constructed for each fault class. Moreover, when studying the diagnosability wrt a given fault class, all fault transitions apart from those that belong to the class at hand, should be considered as regular but unobservable transitions.

Finally, we observe that the problem of finding repetitive sequences that can be enabled after a given transition can be studied in order to look for easier approaches; however this problem is not addressed in this paper.

Example *5.5:* Let us consider again the bounded PN system reported in Fig. 2. Assume that the fault diagnosis is carried out using two local sites as described in Example 5.2.

First, we observe that this system is diagnosable in a centralized framework, e.g., following the approach we proposed in (24). Then, we analyze its diagnosability in a decentralized framework using Proposition 5.4. To this aim we consider the MVN reported in Fig. 3.

By looking at the reachability graph of the MVN, that is not reported here for the sake of brevity, we state that the system has failure ambiguous strings when the two sites are defined as in Example 5.2. In particular, it can be shown that all paths starting from the initial marking and containing the fault t_f end in a cycle labeled c. ■

Note that the approach here presented can be applied to the centralized case when considering $\nu = 1$ and labeling function $\mathcal{L}^j = \bar{\mathcal{L}}$. In fact, it is perfectly equivalent to the approach we proposed in (22) for the centralized case.

VI. Conclusions

In this paper we focused on the problem of analyzing the diagnosability of Petri net systems in the case of diagnosis carried out by a set of local sites and a central coordinator, defined according to a given architecture.

We proved that, regardless of the protocol the different sites use to exchange information with the coordinator and/or among them, the property of diagnosability is strictly related to the presence of particular strings, called failure ambiguous strings. A procedure to detect the presence of failure ambiguous strings is proposed based on the construction of a particular net, denoted as Modified Verifier Net, and on the analysis of cycles associated to repetitive sequences after the occurrence of fault transitions on such a net.

References

[1] F. Lin, "Diagnosability of discrete event systems and its applications," *Discrete Event Dynamic Systems*, vol. 4, no. 2, pp. 197–212, 1994.

[2] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Diagnosability of discrete-event systems," *IEEE Trans. Automatic Control*, vol. 40 (9), pp. 1555–1575, 1995.

[3] R. Boel and J. van Schuppen, "Decentralized failure diagnosis for discrete-event systems with costly communication between diagnosers," in *Proc. WODES'02: 6th Work. on Discrete Event Systems (Zaragoza, Spain)*, Oct. 2002, pp. 175–181.

[4] S. H. Zad, R. Kwong, and W. Wonham, "Fault diagnosis in discrete-event systems: framework and model reduction," *IEEE Trans. Automatic Control*, vol. 48, no. 7, pp. 1199–1212, Jul. 2003.

[5] J. Lunze and J. Schroder, "Sensor and actuator fault diagnosis of systems with discrete inputs and outputs," *IEEE Trans. Systems, Man and Cybernetics, Part B*, vol. 34, no. 3, pp. 1096–1107, Apr. 2004.

[6] T. Ushio, L. Onishi, and K. Okuda, "Fault detection based on Petri net models with faulty behaviors," in *Proc. SMC'98: IEEE Int. Conf. on Systems, Man, and Cybernetics (San Diego, CA, USA)*, Oct. 1998, pp. 113–118.

[7] C. Hadjicostis and G. Veghese, "Monitoring discrete event systems using Petri net embeddings," *Lecture Notes in Computer Science*, vol. 1639, pp. 188–207, 1999.

[8] S. Chung, “Diagnosing pn-based models with partial observable transitions,” *International Journal of Computer Integrated Manufacturing*, vol. 12 (2), pp. 158–169, 2005.

[9] F. Basile, P. Chiacchio, and G. D. Tommasi, “An efficient approach for online diagnosis of discrete event systems,” *IEEE Trans. Automatic Control*, vol. 54, no. 4, pp. 748–759, 2008.

[10] M. Dotoli, M. Fanti, and A. Mangini, “Fault detection of discrete event systems using Petri nets and integer linear programming,” in *Proc. of 17th IFAC World Congress*, Seoul, Korea, Jul. 2008.

[11] R. Debouk, S. Lafortune, and D. Teneketzis, “Coordinated decentralized protocols for failure diagnosis of discrete-event systems,” *Discrete Events Dynamic Systems*, vol. 10, no. 1, pp. 33–86, 2000.

[12] R. Su, W. Wonham, J. Kurien, and X. Koutsoukos, “Distributed diagnosis for qualitative systems,” in *in 6th International Workshop on Discrete Event Systems, Zaragoza*, 2002, pp. 169–174.

[13] O. Contant, S. Lafortune, and D. Teneketzis, “Diagnosability of discrete event systems with modular structure,” *Discrete Event Dynamic Systems*, vol. 16, no. 1, pp. 9–37, 2006.

[14] W. Qiu and R. Kumar, “Decentralized failure diagnosis of discrete event systems,” *IEEE Trans. Systems, Man and Cybernetics, Part A*, vol. 36, no. 2, 2006.

[15] Y. Wang, T.-S. Yoo, and S. Lafortune, “Diagnosis of discrete event systems using decentralized architectures,” *Discrete Event Dynamic Systems*, vol. 17, no. 2, 2007.

[16] M. Moreira, T. Jesus, and J. Basilio, “Polynomial time verification of decentralized diagnosability of discrete event systems,” in *American Control Conference (ACC), 2010*, Jun. 2010, pp. 3353–3358.

[17] A. Benveniste, E. Fabre, S. Haar, and C. Jard, “Diagnosis of asynchronous discrete event systems, a net unfolding approach,” *IEEE Trans. Automatic Control*, vol. 48, no. 5, pp. 714–727, May 2003.

[18] G. Jiroveanu and R. K. Boel, “A distributed approach for fault detection and diagnosis based on time Petri nets,” *Mathematics and Computers in Simulation*, vol. 70, no. 5, 2006.

[19] S. Genc and S. Lafortune, “Distributed diagnosis of place-bordered Petri nets,” *IEEE Trans. on Automation Science and Engineering*, vol. 4, no. 2, pp. 206–219, 2007.

[20] M. Cabasino, A. Giua, A. Paoli, and C. Seatzu, “A new protocol for the decentralized

diagnosis of labeled Petri nets," in *10th Workshop on Discrete Event Systems*, Berlin, Germany, aug-sep 2010.

[21] ——, "Decentralized diagnosis of Petri nets," in *Proc. 2010 American Control Conference*, 2010.

[22] M. Cabasino, A. Giua, S. Lafortune, and C. Seatzu, "Diagnosability analysis of unbounded Petri nets," in *Proc. 48th IEEE Conf. on Decision and Control*, Dec. 2009.

[23] C. Cassandras and S. Lafortune, *Introduction to discrete event systems, Second Edition*. Springer, 2007.

[24] M. Cabasino, A. Giua, and C. Seatzu, "Diagnosability of bounded Petri nets," in *Proc. 48th IEEE Conf. on Decision and Control*, Dec. 2009.