An Improved Approach to Test Diagnosability of Bounded Petri Nets

Ning Ran, Hongye Su, *Senior Member, IEEE*, and Shouguang Wang, *Senior Member, IEEE*

***Abstract*—For bounded Petri nets, Cabasino *et al.* propose a diagnosability test method that is based on the analysis of a modified basis reachability graph and a basis reachability diagnoser. However, its complexity is exponential in the number of nodes of the basis reachability diagnoser. In order to reduce the complexity of their method, this paper presents a new diagnosability test approach for bounded Petri nets. We present the concept of an extended basis reachability graph and prove that our approach is of polynomial complexity in the number of nodes of extended basis reachability graphs. An example is given to illustrate the application of the presented approach.**

***Index Terms*—Diagnosability, discrete event systems (DES), fault diagnosis, Petri nets.**

I. Introduction

FAULT diagnosis is an important task in complex and large systems. In a discrete event system (DES) framework, diagnosability is a required property for many practical applications. Diagnosability problem has received much attention since the early 1990s [1]−[12]. Solving a diagnosability problem is equivalent to determining if, once a fault has occurred, the occurrence of the fault can be detected with an observation of bounded length. Sampath *et al.* [1] first formally presented the definition of diagnosability for regular languages and proposed a method to test diagnosability of DESs modeled by automata.

Due to the intuitive graphical representation and powerful algebraic formulation, Petri nets have been recently used to deal with diagnosability problems and a series of diagnosability methods have been developed [5]−[12]. Ushio *et al.* [7] extend the method for automata in [1] to test diagnosability of unbounded Petri nets under the assumption that all transitions are unobservable and places are partitioned into observable and unobservable ones. They construct a so-called simple ω diagnoser based on the coverability tree and present a sufficient condition for diagnosability of a net.

Manuscript received August 28, 2016; accepted September 30, 2016. This work was supported by National Natural Science Foundation of China (61134007), National Basic Research Program of China (2013CB035406). Recommended by Associate Editor Mengchu Zhou.

Citation: N. Ran, H. Y. Su, and S. G. Wang, "An improved approach to test diagnosability of bounded petri nets," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 2, pp. 297−303, Apr. 2017.

N. Ran and H. Y. Su are with the State Laboratory of Industrial Control Technology, Institute of Cyber-Systems and Control, Zhejiang University, Yuquan Campus, Hangzhou 310027, China (e-mail: ranning87@hotmail.com; hysu@iipc.zju.edu.cn).

S. G. Wang is with the School of Information & Electronic Engineering, Zhejiang Gongshang University, Hangzhou 310018, China (e-mail: wsg5000@hotmail.com).

Digital Object Identifier 10.1109/JAS.2017.7510406

Chung [8] generalizes the setting in [7] by assuming that some of the transitions are observable, and constructs a diagnoser by taking the advantage of the additional information from observable transitions. Then a verifier is proposed to test diagnosability of bounded Petri nets.

Wen *et al.* [9] test diagnosability of Petri nets by checking their T-invariants. They employ the diagnoser proposed in [7] to prove that their method is correct. However they do not actually construct a diagnoser to test diagnosability. Wen and Jeng [10] then proposed a diagnosability test algorithm by using a linear programming. It is of polynomial complexity in the number of net nodes and provides a sufficient condition for diagnosability of the nets.

Cabasino *et al.* [11] give a necessary and sufficient condition for diagnosability of both bounded and unbounded Petri nets and then present an algorithm for testing the condition based on linear programming. Their approach is a general technique and based on the analysis of the reachability/coverability graph of a Petri net model named a verifier net. However, the number of the states of the reachability graph increases exponentially with the system complexity.

Cabasino *et al.* [12] compute basis markings of bounded Petri nets to overcome the problem of exhaustive enumeration of the state space. The diagnosability test method is based on the analysis of two deterministic graphs called a basis reachability graph and a basis reachability diagnoser. A Petri net is diagnosable if and only if its basis reachability diagnoser has no cycle that is indeterminate wrt all fault classes. However, the complexity of their method is exponential in the number of nodes in the basis reachability diagnoser.

In order to address the complexity problem in [12], we propose a new diagnosability approach for bounded Petri nets. We propose and compute an extended basis reachability graph for each fault class and construct verifiers based on these graphs. A necessary and sufficient condition is presented for diagnosability of bounded Petri nets. The proposed method is of polynomial complexity in the number of nodes of the extended basis reachability graphs.

The rest of this paper is organized as follows. Section II briefly reviews preliminaries used in this paper. The notion of diagnosability of a Petri net is introduced in Section III. Section IV proposes some new definitions and an approach to test diagnosability of Petri nets. Section V presents an example to demonstrate the proposed approach. Section VI compares our approach and the one proposed in [11] from the complexity point of view. Finally, Section VII concludes this paper.

II. Preliminaries

In this section, basic definitions of Petri nets are reviewed [13]−[16].

A Petri net 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 flow relation, $W : (P \times T) \cup (T \times P)$ is a mapping that assigns a weight to each arc: $W(x, y) > 0$ if $(x, y) \in F$, and $W(x, y) = 0$ otherwise, where $x, y \in P \cup T$.

Let $x \in P \cup T$ be a node of net $N = (P, T, F, W)$. The preset of x is ${}^\bullet x = \{y \in P \cup T | (y, x) \in F\}$, and the postset of x is $x^\bullet = \{y \in P \cup T | (x, y) \in F\}$. The incidence matrix $[N]$ of N is a $|P| \times |T|$ integer matrix with $[N](p, t) = W(t, p) - W(p, t)$. A Petri net that has no directed circuits is said to be acyclic.

A marking m is a mapping from P to $\mathbb{N} = \{0, 1, 2, \ldots\}$. (N, m_0) denotes a Petri net with an initial marking m_0. $m(p)$ denotes the number of tokens in place p. A transition $t \in T$ is enabled at a marking m if $\forall p \in {}^\bullet t, m(p) \geq W(p, t)$, which is denoted by $m[t >$. Firing t yields a new marking m': $\forall p \in P, m'(p) = m(p) + [N](p, t)$, which is denoted by $m[t > m'$. A marking m'' is said to be reachable from m if there exists a transition sequence σ such that $m[\sigma > m''$. The set of markings reachable from m in N is called the reachability set of a marked Petri net (N, m) and denoted by $R(N, m)$. A Petri net is said to be bounded if there exists a positive constant k such that $\forall m \in R(N, m_0), \forall p \in P, m(p) \leq k$.

The set of all sequences that are enabled at the initial marking m_0 is denoted by $L(N, m_0)$, i.e., $L(N, m_0) = \{\sigma \in T^* | m_0[\sigma >\}$. ε is used to denote the empty sequence. Let σ be a transition sequence. Its Parikh vector is denoted by $\pi(\sigma)$. We use $t \in \sigma$ to denote that a transition t is contained in σ. Moreover, we use $T' \in \sigma$ to denote that there exists at least one transition in $T' \subset T$ contained in σ, and $T' \notin \sigma$ to denote that there is no transition in T' contained in σ.

Given a Petri net $N = (P, T, F, W)$ and a set $T' \subseteq T$ of transitions, we define T'-induced subnet of N as a new Petri net $N' = (P, T', F', W)$, where F' is the restriction of F to $(P \times T') \cup (T' \times P)$.

A labeled Petri net is a triple $(N, m_0, \mathcal{L})$, where (N, m_0) is a marked Petri net, and $\mathcal{L}$ is a labeling function $\mathcal{L} : T \to L \cup \{\varepsilon\}$ that assigns to each transition in T either a symbol from a given alphabet L or ε. Given a transition sequence $\sigma = t_1 t_2 \cdots t_k \in T^*$, the corresponding observation is $w = \mathcal{L}(\sigma) = \mathcal{L}(t_1)\mathcal{L}(t_2)\cdots\mathcal{L}(t_k)$. $\mathcal{L}^{-1}(w)$ denotes the set of all transition sequences consistent with $w \in L^*$, i.e., $\mathcal{L}^{-1}(w) = \{\sigma \in L(N, m_0) | \mathcal{L}(\sigma) = w\}$.

We use T_u to denote the set of transitions whose labels are ε, and T_o to denote the set of transitions whose labels are the symbols in L. T_u and T_o are called the set of unobservable and observable transitions, respectively. $[N]_u$ (or $[N]_o$) is used to denote the restriction of the incidence matrix $[N]$ to T_u (or T_o).

Given a transition sequence $\sigma \in T^*$, we denote $P_u(\sigma)$ (or $P_o(\sigma)$) as the projection of σ over T_u (or T_o). Let $K \subseteq T^*$ be a language, we use K/σ to denote the post-language of K after σ, i.e., $K/\sigma = \{\sigma' | \sigma\sigma' \in K\}$.

III. Problem Statement

We partition the unobservable transition set as $T_u = T_f \cup T_r$, where T_f is the set of faults and T_r is the set of unobservable but regular transitions. All faults in T_f are partitioned into r different subsets T_f^i that model different fault classes, where $i = 1, 2, \ldots, r$. In the following, we use T_r^i to denote the set of unobservable transitions that does not contain the faults in T_f^i, i.e., $T_r^i = T_u \setminus T_f^i$, and $[N]_r^i$ to denote the restriction of the incidence matrix to T_r^i.

We make the following assumptions in this paper, which are the same as the assumptions in [12]:

A1) The T_u-induced subnet is acyclic.

A2) The Petri net is deadlock-free.

A3) The Petri net is bounded.

Note that Assumption A1 is commonly adopted in the diagnosability study of Petri nets. It is analogous to the classical assumption in the framework of automata that there does not exist any cycle of unobservable events [1].

Just like the problem of model identification of DESs [17], the problem of fault diagnosis of DESs also belongs to the identification problem. The former aims to decide whether for the given behavioral specification there exists a DES that generates the specified behavior, and provide a constructive procedure to determine such a DES [17]. The latter focuses on detecting whether a fault event has occurred according to the behavior generated by a DES.

Now we introduce the definition of diagnosability of labeled Petri nets.

We use $\Psi(T_f^i)$ to denote the set of all sequences in $L(N, m_0)$ that end with a transition in T_f^i.

Definition 1 [12]: Given a deadlock-free labeled Petri net $(N, m_0, \mathcal{L})$, the i-th fault class T_f^i is said to be diagnosable if $L(N, m_0)$ is diagnosable wrt $\mathcal{L}$ and T_f^i, i.e.,

$$\forall \sigma' \in \Psi(T_f^i), \quad \exists k \in \mathbb{N}, \quad \forall \sigma'' \in \frac{L(N, m_0)}{\sigma'}$$

$$|\sigma''| \geq k \Rightarrow D$$

where the diagnosability condition D is

$$\forall \sigma \in \mathcal{L}^{-1}(\mathcal{L}(\sigma'\sigma'')), \exists t_f \in T_f^i \Rightarrow t_f \in \sigma.$$

A deadlock-free labeled Petri net $(N, m_0, \mathcal{L})$ is said to be diagnosable if all fault classes are diagnosable.

In simple words, T_f^i is diagnosable if, once a fault in T_f^i has occurred, the system can detect the occurrence of a fault belonging to T_f^i within a finite delay.

IV. Main Results

A. Extended Basis Reachability Graph

Similar to [12], this work computes the basis markings of a Petri net to avoid enumerating all states. The following notions are used.

Definition 2 [12]: Given a marking m and an observable transition t, the set of explanations of t at m is denoted by

$$\Sigma(m, t) = \{\sigma \in T_u^* | m[\sigma > m', m'[t >\}$$

and the set of e-vectors is denoted by

$$Y(m,t) = \pi(\Sigma(m,t)).$$

Definition 3 [12]: Given a marking m and an observable transition t, the set of minimal explanations of t at m is denoted by

$$\Sigma_{\min}(m,t) = \{\sigma \in \Sigma(m,t) | \nexists \sigma' \in \Sigma(m,t) : \pi(\sigma') \lneq \pi(\sigma)\}$$

and the set of minimal e-vectors is denoted by

$$Y_{\min}(m,t) = \pi(\Sigma_{\min}(m,t)).$$

Definition 4 [12]: Let $(N, m_0, \mathcal{L})$ be a labeled Petri net and $w \in L^*$ be an observation, where $N = (P, T, F, W)$ and $T = T_o \cup T_u$. The set of pairs ($\sigma_o \in T_o^*$ with $\mathcal{L}(\sigma_o) = w$ and the justification) is denoted by

$$\begin{aligned}\hat{\mathcal{J}}(w) = \{&(\sigma_o, \sigma_u), \sigma_o \in T_o^*, \mathcal{L}(\sigma_o) = w, \sigma_u \in T_u^* | \\ &[\exists \sigma \in \mathcal{L}^{-1}(w) : \sigma_o = P_o(\sigma), \sigma_u = P_u(\sigma)] \\ &\wedge [\nexists \sigma' \in \mathcal{L}^{-1}(w) : \sigma_o = P_o(\sigma'), \\ &\sigma_u' = P_u(\sigma') \wedge \pi(\sigma_u') \lneq \pi(\sigma_u)]\}\end{aligned}$$

and the set of pairs ($\sigma_o \in T_o^*$ with $\mathcal{L}(\sigma_o) = w$ and the j-vector) is denoted by

$$\begin{aligned}\hat{Y}_{\min}(m_0, w) = \{&(\sigma_o, y), \sigma_o \in T_o^*, \mathcal{L}(\sigma_o) = w, y \in \mathbb{N}^{|T_u|} | \\ &\exists (\sigma_o, \sigma_u) \in \hat{\mathcal{J}}(w) : \pi(\sigma_u) = y\}.\end{aligned}$$

Definition 5 [12]: Given a pair $(\sigma_o, \sigma_u) \in \hat{\mathcal{J}}(w)$, the basis marking is denoted by

$$m_b = m_0 + [N]_u \cdot \pi(\sigma_u) + [N]_o \cdot \pi(\sigma_o).$$

Cabasino *et al.* [12] construct a basis reachability graph (BRG) by assuming that all faults in T_f are observable, while we construct a BRG for each fault class by assuming that the faults in this class are observable. In more detail, when we construct the BRG for a given fault class T_f^i, the faults in T_f^i are considered as observable transitions and the faults in $T_f \setminus T_f^i$ are considered as regular unobservable transitions. We provide the following definitions that are modified according to the ones presented in [12].

Definition 6: An extended basis marking (EBM) wrt T_f^i is a basis marking computed by assuming that all faults in T_f^i are observable. The set of all EBMs wrt T_f^i is denoted by M_e^i.

M_e^i can be computed by restricting the minimal explanations to the unobservable transitions within T_r^i. In the following, we denote $Y_{\min}^i(m,t)$ as the set of minimal e-vectors restricted to T_r^i. $Y_{\min}^i(m,t)$ can be computed by Algorithm 4.4 presented in [15].

Definition 7: An extended basis reachability graph (EBRG) wrt T_f^i is a (non-deterministic) graph that has as many nodes as the number of markings in M_e^i. Each node is associated a different marking in M_e^i. Each arc is labeled with t_f or $e(t)$, where $t_f \in T_f^i$, $e \in L$ and $t \in T_o$ is the transition labeled with e. More precisely, an arc t_f (or $e(t)$) exists from nodes m to m' if there exists a minimal e-vector $y \in Y_{\min}^i(m, t_f)$ (or $y \in Y_{\min}^i(m,t)$) that satisfies $m' = m + [N]_r^i \cdot y + [N](\cdot, t_f)$ (or $m' = m + [N]_r^i \cdot y + [N](\cdot, t)$).

In the following, we denote G_E^i as the EBRG wrt T_f^i. Given a labeled Petri net $(N, m_0, \mathcal{L})$ that satisfies Assumptions A1−A3, Algorithm 1 summarizes the main steps for the construction of G_E^i.

Algorithm 1 Construction of the EBRG wrt T_f^i

Input: $(N, m_0, \mathcal{L})$ and T_f^i.
Output: G_E^i.
1. Let m_0 be the initial node and tag it "new".
2. While nodes with "new" exist
  2.1. select a node m with "new",
  2.2. for all $t \in T_o \cup T_f^i$, do
    2.2.1. if $Y_{\min}^i(m,t) \neq \emptyset$, then
      for all $y \in Y_{\min}^i(m,t)$, do
        let $m' = m + [N]_r^i \cdot y + [N](\cdot, t)$,
        if $\nexists$ a node m', then
          add a node m' and tag it "new",
        if $t \in T_o \wedge \nexists$ an arc $e(t)$ from m to m', where $e = \mathcal{L}(t)$, then
          add an arc $e(t)$ from m to m',
        if $t \in T_f^i \wedge \nexists$ an arc t from m to m', then
          add an arc t from m to m',
  2.3. remove the tag of m.

Consider the labeled Petri net shown in Fig. 1, where $T_o = \{t_2, t_5, t_6\}$, $T_u = \{t_1, t_3, t_4\}$, $T_f = \{t_4\}$ and $m_0 = [2\ 1\ 0\ 0\ 0\ 0\ 0]^T$. The labeling function is defined as follows: $\mathcal{L}(t_2) = a$ and $\mathcal{L}(t_5) = \mathcal{L}(t_6) = b$. The corresponding EBMs wrt T_f are detailed in Table I and the EBRG wrt T_f is shown in Fig. 2.

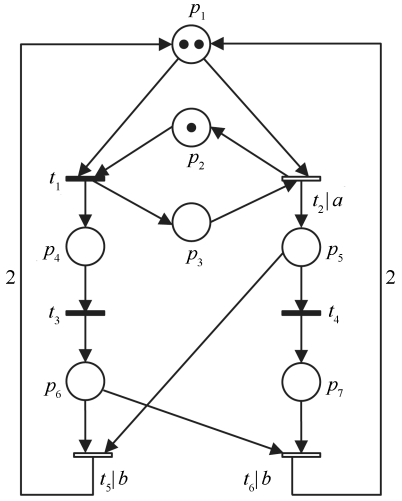

Fig. 1. A labeled Petri net.

TABLE I
EBMs WRT T_f

Node	EBM
m_0	$[2\ 1\ 0\ 0\ 0\ 0\ 0]^T$
m_1	$[0\ 1\ 0\ 1\ 1\ 0\ 0]^T$
m_2	$[0\ 1\ 0\ 1\ 0\ 0\ 1]^T$

For each path starting from the initial node of G_E^i, we build a sequence such that for each arc in the path we take either t_f if the arc is labeled by t_f or t if the arc is labeled by $e(t)$. The set of all sequences built in this way is denoted by $L_E^i(N, m_0)$.

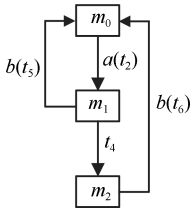

Fig. 2. EBRG wrt T_f.

From Algorithm 1 and Assumption A1, we can immediately derive the following two properties.

Property 1: Let σ be a sequence of infinite length in $L(N, m_0)$. If σ contains a fault $t_f \in T_f^i$, then there exists a sequence of infinite length $\sigma' \in L_E^i(N, m_0)$ that contains t_f and satisfies $\mathcal{L}(\sigma') = \mathcal{L}(\sigma)$, and vice versa.

Proof: According to Algorithm 1, the set of transition sequences in $L_E^i(N, m_0)$ coincides with the projection of $L(N, m_0)$ over the set $T_o \cup T_f^i$. Therefore σ contains a fault $t_f \in T_f^i$ iff σ' contains a fault $t_f \in T_f^i$, where $\mathcal{L}(\sigma') = \mathcal{L}(\sigma)$. Moreover, by Assumption A1 σ is infinitely long iff σ' is infinitely long. Hence, the result holds. ■

Property 2: Let σ be a sequence of infinite length in $L(N, m_0)$. If σ does not contain a fault in T_f^i, then there exists a sequence of infinite length $\sigma' \in L_E^i(N, m_0)$ that does not contain a fault in T_f^i and satisfies $\mathcal{L}(\sigma') = \mathcal{L}(\sigma)$, and vice versa.

Proof: The proof follows the same lines of the proof of Property 1 and is omitted. ■

Lemma 1: Given a labeled Petri net $(N, m_0, \mathcal{L})$ that satisfies Assumptions A1−A3, a fault class T_f^i is diagnosable iff $L_E^i(N, m_0)$ is diagnosable wrt $\mathcal{L}$ and T_f^i.

Proof: (If) By contradiction, suppose that T_f^i is not diagnosable. According to Definition 1, there must exist two sequences of infinite length $\sigma_1, \sigma_2 \in L(N, m_0)$ such that $\mathcal{L}(\sigma_1) = \mathcal{L}(\sigma_2)$, $T_f^i \in \sigma_1$ and $T_f^i \notin \sigma_2$. From Properties 1 and 2, we know that there exist two sequences of infinite length $\sigma_1', \sigma_2' \in L_E^i(N, m_0)$ such that $\mathcal{L}(\sigma_1') = \mathcal{L}(\sigma_2') = \mathcal{L}(\sigma_1) = \mathcal{L}(\sigma_2)$, $T_f^i \in \sigma_1'$ and $T_f^i \notin \sigma_2'$. Hence $L_E^i(N, m_0)$ is not diagnosable wrt $\mathcal{L}$ and T_f^i. This is a contradiction.

(Only if) By contradiction, suppose that $L_E^i(N, m_0)$ is not diagnosable wrt $\mathcal{L}$ and T_f^i. According to Definition 1, there must exist two sequences of infinite length $\sigma_1', \sigma_2' \in L_E^i(N, m_0)$ such that $\mathcal{L}(\sigma_1') = \mathcal{L}(\sigma_2')$, $T_f^i \in \sigma_1'$ and $T_f^i \notin \sigma_2'$. From Properties 1 and 2, we know that there exist two sequences of infinite length $\sigma_1, \sigma_2 \in L(N, m_0)$ such that $\mathcal{L}(\sigma_1) = \mathcal{L}(\sigma_2) = \mathcal{L}(\sigma_1') = \mathcal{L}(\sigma_2')$, $T_f^i \in \sigma_1$ and $T_f^i \notin \sigma_2$. Hence T_f^i is not diagnosable. This is a contradiction. ■

B. Extended Observer

Next, we present the notion of an extended observer, which can be constructed from an EBRG.

Definition 8: An extended observer (EO) wrt T_f^i is a (nondeterministic) graph. Each node is labeled with (m, h), where m is a basis marking, $h \in \{N, F\}$. In particular, $h = N$ if, during the evolution from m_0 to m, no fault in T_f^i has occurred and $h = F$ otherwise. Arcs are labeled with symbols in L.

In the following, we denote G_O^i as the EO wrt T_f^i. Algorithm 2 summarizes the main steps for the construction of G_O^i.

Algorithm 2 Construction of the EO wrt T_f^i

Input: G_E^i.
Output: G_O^i.
1. Let (m_0, N) be the initial node and tag it "new".
2. While nodes with "new" exist
  2.1. select a node $d = (m, h)$ with "new",
  2.2. for all $t \in T_o$, do
    2.2.1. if $e(t)$ is an output arc of m in G_E^i, then
      let m' be the output node in G_E^i and $h' = h$,
      if $\nexists$ a node $d' = (m', h')$, then
        add a node d' and tag it "new",
      if $\nexists$ an arc e from d and d', then
        add an arc e from d to d',
    2.2.2. for all out paths of m in G_E^i lebeled $\sigma e(t)$, where $\sigma \in (T_f^i)^*$ and $\sigma \neq \varepsilon$, do
      let m'' be the final node in G_E^i and $h'' = F$,
      if $\nexists$ a node $d'' = (m'', h'')$, then
        add a node d'' and tag it "new",
      if $\nexists$ an arc e from d and d'', then
        add an arc e from d to d'',
  2.3. remove the tag of d.

Since there may exist more than one path from m_0 to m such that some contain faults and others not, m may appear in two different nodes: (m, N) and (m, F). According to Algorithm 2 and Assumption A1, we can derive the following properties.

Property 3: Given a cycle

$$c = ((m_1, h_1), e_1, (m_2, h_2), \ldots, (m_k, h_k), e_k, (m_1, h_1))$$

in G_O^i, we have $h_a = h_b$ for any a and b in $\{1, 2, \ldots, k\}$.

Proof: Straightforward from Step 2.2 of Algorithm 2. ■

Property 4: Let σ be a sequence of infinite length in $L_E^i(N, m_0)$. If σ does not contain a fault in T_f^i, then there exists a path λ ending with a cycle in G_O^i, where

$$\lambda = ((m_0, h_0), e_0, \ldots, (m_{k-1}, h_{k-1}), e_{k-1}, (m_k, h_k), e_k, \ldots, (m_n, h_k), e_n, (m_k, h_k))$$

such that: 1) $\mathcal{L}(\sigma) = (e_0, e_1, \ldots, e_{k-1}, (e_k, \ldots, e_n)^\infty)$, and 2) $h_k = N$, and vice versa.

Proof: Since the Petri net is bounded by Assumption A3, the path λ obviously ends with a cycle, i.e., $\mathcal{L}(\sigma) = (e_0, e_1, \ldots, e_{k-1}, (e_k, \ldots, e_n)^\infty)$. Moreover, σ does not contain a fault in T_f^i iff the sequence $\mathcal{L}(\sigma)$ does not contain a fault in T_f^i by Property 2. Hence, it holds that $h_k = N$.

Similarly, we can prove that if the path λ is ending with a cycle such that $\mathcal{L}(\sigma) = (e_0, e_1, \ldots, e_{k-1}, (e_k, \ldots, e_n)^\infty)$ and $h_k = N$, then there exists an infinitely long sequence σ that does not contain a fault in T_f^i. ■

Property 5: Let σ be a sequence of infinite length in $L_E^i(N, m_0)$. If σ contains a fault $t_f \in T_f^i$, then there exists a path λ ending with a cycle in G_O^i, where

$$\lambda = ((m_0, h_0), e_0, \ldots, (m_{k-1}, h_{k-1}), e_{k-1}, (m_k, h_k), e_k, \ldots, (m_n, h_k), e_n, (m_k, h_k))$$

such that: 1) $\mathcal{L}(\sigma) = (e_0, e_1, \ldots, e_{k-1}, (e_k, \ldots, e_n)^\infty)$, and 2) $h_k = F$, and vice versa.

Proof: The proof follows the same lines of the proof of Property 4 and is omitted. ■

Lemma 2: Given a labeled Petri net $(N, m_0, \mathcal{L})$ that satisfies Assumptions A1−A3, T_f^i is diagnosable iff there do not exist two paths λ_1 and λ_2 ending with cycles in G_O^i, where

$$\lambda_1 = ((m_0^1, h_0^1), e_0, \ldots, (m_{k-1}^1, h_{k-1}^1), e_{k-1}, (m_k^1, h_k^1), e_k, \ldots, (m_n^1, h_k^1), e_n, (m_k^1, h_k^1)) \text{ and}$$

$$\lambda_2 = ((m_0^2, h_0^2), e_0, \ldots, (m_{k-1}^2, h_{k-1}^2), e_{k-1}, (m_k^2, h_k^2), e_k, \ldots, (m_n^2, h_k^2), e_n, (m_k^2, h_k^2))$$

such that $h_k^1 \neq h_k^2$.

Proof: (If) By contradiction, suppose that T_f^i is not diagnosable. According to Lemma 1 and Definition 1, there must exist two sequences of infinite length $\sigma_1, \sigma_2 \in L_E^i(N, m_0)$ such that: $\mathcal{L}(\sigma_1) = \mathcal{L}(\sigma_2)$, $T_f^i \in \sigma_1$ and $T_f^i \notin \sigma_2$. From Properties 3, 4 and 5, we know that there exist two paths λ_1 and λ_2 in G_O^i, where

$$\lambda_1 = ((m_0^1, h_0^1), e_0, \ldots, (m_{k-1}^1, h_{k-1}^1), e_{k-1}, (m_k^1, h_k^1), e_k, \ldots, (m_n^1, h_k^1), e_n, (m_k^1, h_k^1)) \text{ and}$$

$$\lambda_2 = ((m_0^2, h_0^2), e_0, \ldots, (m_{k-1}^2, h_{k-1}^2), e_{k-1}, (m_k^2, h_k^2), e_k, \ldots, (m_n^2, h_k^2), e_n, (m_k^2, h_k^2))$$

such that: 1) $(e_0, e_1, \ldots, e_{k-1}, (e_k, \ldots, e_n)^\infty) = \mathcal{L}(\sigma_1) = \mathcal{L}(\sigma_2)$, and 2) $h_k^1 \neq h_k^2$. This is a contradiction.

(Only if) By contradiction, suppose that there exist two paths λ_1 and λ_2 in G_O^i, where

$$\lambda_1 = ((m_0^1, h_0^1), e_0, \ldots, (m_{k-1}^1, h_{k-1}^1), e_{k-1}, (m_k^1, h_k^1), e_k, \ldots, (m_n^1, h_k^1), e_n, (m_k^1, h_k^1)), \text{ and}$$

$$\lambda_2 = ((m_0^2, h_0^2), e_0, \ldots, (m_{k-1}^2, h_{k-1}^2), e_{k-1}, (m_k^2, h_k^2), e_k, \ldots, (m_n^2, h_k^2), e_n, (m_k^2, h_k^2))$$

such that: $h_k^1 \neq h_k^2$. According to Properties 4 and 5, there must exist two sequences of infinite length $\sigma_1, \sigma_2 \in L_E^i(N, m_0)$ such that $\mathcal{L}(\sigma_1) = \mathcal{L}(\sigma_2) = (e_0, e_1, \ldots, e_{k-1}, (e_k, \ldots, e_n)^\infty)$, $T_f^i \in \sigma_1$ and $T_f^i \notin \sigma_2$. From Lemma 1, we know that T_f^i is not diagnosable. This is a contradiction. ■

Consider the EBRG wrt T_f shown in Fig. 2. According to Algorithm 2, we construct the the EO wrt T_f, which is shown in Fig. 3.

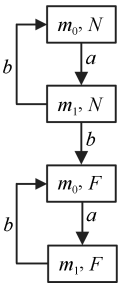

Fig. 3. Extended observer wrt T_f.

Remark 1: We can test diagnosability of a labeled Petri net by examining all cycles and the paths that lead from the initial node to each cycle in G_O^i, $\forall i \in \{1, 2, \ldots, r\}$. However, this test is analogous to the method in [7], and thus its complexity is exponential in the number of nodes in G_O^i. In the next subsection, an approach with polynomial complexity is developed.

C. Verifier

In this subsection, we propose an approach to test diagnosability of a labeled Petri net satisfying Assumptions A1−A3 by analyzing the verifier of each fault class respectively.

Here we present the definition of a verifier.

Definition 9: A verifier wrt T_f^i is a (non-deterministic) graph. Each node is labeled with $(d_1; d_2)$, where d_1 and d_2 are nodes in G_O^i. Arcs are labeled with symbols in L. More precisely, an arc e exists from a node $(d_1; d_2)$ to a node $(d_1'; d_2')$ if in G_O^i there exists an arc labeled e from d_1 to d_2 and an arc labeled e from d_1' to d_2'.

In the following, we use G_V^i to denote the verifier wrt T_f^i. Algorithm 3 summarizes the main steps for the construction of G_V^i.

Algorithm 3 Construction of the verifier wrt T_f^i

Input: G_O^i.
Output: G_V^i.
1. Let $(m_0, N; m_0, N)$ be the initial node and tag it "new".
2. While nodes with "new" exist
  2.1. select a node $d = (m_1, h_1; m_2, h_2)$ with "new",
  2.2. for all $e \in L$, do
    2.2.1. if (m_1, h_1) and (m_2, h_2) both have output arcs labeled e in G_O^i, then
      let M be the set of all output nodes in G_O^i,
      for all $(m_1', h_1') \in M$, do
        for all $(m_2', h_2') \in M$, do
          if $h_1' = N \vee h_2' = N$
            if ($\nexists$ a node $d' = (m_1', h_1'; m_2', h_2')) \wedge$
            ($\nexists$ a node $d'' = (m_2', h_2'; m_2', h_1')$)
              add a node d' and tag it "new",
              add an arc e from d to d',
  2.3. remove the tag of d.

From Algorithm 3, we can directly derive the following lemma.

Lemma 3: Let λ be a path in G_V^i ending with a cycle, where

$$\lambda = (x_0, e_0, \ldots, x_k, e_k, \ldots, x_n, e_n, x_k),$$

$$x_j = (m_j^1, h_j^1; m_j^2, h_j^2), j = 0, 1, 2, \ldots, n$$

the following holds:

There exist two paths λ_1 and λ_2 ending with cycles in G_O^i, where

$$\lambda_1 = ((m_0^1, h_0^1), e_0, \ldots, (m_{k-1}^1, h_{k-1}^1), e_{k-1}, (m_k^1, h_k^1), e_k, \ldots, (m_n^1, h_k^1), e_n, (m_k^1, h_k^1)), \text{ and}$$

$$\lambda_2 = ((m_0^2, h_0^2), e_0, \ldots, (m_{k-1}^2, h_{k-1}^2), e_{k-1}, (m_k^2, h_k^2), e_k, \ldots, (m_n^2, h_k^2), e_n, (m_k^2, h_k^2)).$$

Next we present a necessary and sufficient condition for diagnosability of a labeled Petri net.

Theorem 1: Given a labeled Petri net $(N, m_0, \mathcal{L})$ that satisfies Assumptions A1−A3, a fault class T_f^i is diagnosable iff for each cycle $c = (x_1, e_1, x_2, \ldots, x_n, e_n, x_1)$ in G_V^i, where $x_j = (m_j^1, h^1; m_j^2, h^2)$ and $j = 1, 2, \ldots, n$, we have $h^1 = h^2$.

Proof: Straightforward from Lemmas 2 and 3. ■

Algorithm 4 summarizes the main steps for testing diagnosability of a labeled Petri net $(N, m_0, \mathcal{L})$ that satisfies Assumptions A1−A3.

Algorithm 4 Diagnosability-Test Algorithm

Input: $(N, m_0, \mathcal{L})$.

Output: Diagnosability of $(N, m_0, \mathcal{L})$.

1. For all T_f^i, $i = \{1, 2, \ldots, r\}$, do

1.1. construct G_E^i by Algorithm 1,

1.2. compute G_O^i from Algorithm 2,

1.3. obtain G_V^i by Algorithm 3,

1.4. check whether there exists in G_V^i a cycle $c = (x_1, e_1, x_2, \ldots, x_n, e_n, x_1)$, $x_j = (m_j^1, h_j^1; m_j^2, h_j^2)$, $j \in \{1, 2, \ldots, n\}$, such that $h_j^1 \neq h_j^2$,

1.4.1. if the answer is yes, then output T_f^i is not diagnosable,

1.4.1. else output T_f^i is diagnosable.

2. If all fault classes are diagnosable, then

2.1. output $(N, m_0, \mathcal{L})$ is diagnosable.

We conclude this subsection with a discussion on the complexity of the proposed approach.

Theorem 2: Let $(N, m_0, \mathcal{L})$ be a labeled Petri net that satisfies Assumptions A1−A3, and x be the maximum number of nodes in G_E^i for all i in $\{0, 1, \ldots, r\}$. The complexity of testing diagnosability of the net is $O(x^4 \times |T| \times r)$.

Proof: From Definition 8, we know that the number of nodes in G_O^i is at most $2x$, and the number of arcs in G_O^i is at most $4x^2 \times |T|$. According to Definition 9, the number of nodes in G_V^i is at most $4x^2$, and the number of arcs in G_V^i is at most $16x^4 \times |T|$. Moreover, we need to check all cycles in G_V^i. This can be computed by Tarjan's strongly connected components algorithm [1], whose complexity is linear in the number of nodes and arcs in G_V^i, i.e., $O(x^4 \times |T|)$. Hence, the overall complexity is $O(x^4 \times |T| \times r)$. ■

V. Example

Let us continue to consider the labeled Petri net in Fig. 1, whose EBRG wrt T_f is shown in Fig. 2 and EBMs wrt T_f are detailed in Table I. Its EO wrt T_f and verifier wrt T_f are shown in Figs. 3 and 4, respectively.

In Fig. 4, there are two cycles (x_1, a, x_2, b, x_1) and (x_1', a, x_2', b, x_1') surrounded by dash lines, where $x_1 = (m_0, N; m_0, F)$, $x_2 = (m_1, N; m_1, F)$, $x_1' = (m_0, F; m_0, N)$ and $x_2' = (m_1, F; m_1, N)$. According to Theorem 1, we know that the labeled Petri net is not diagnosable.

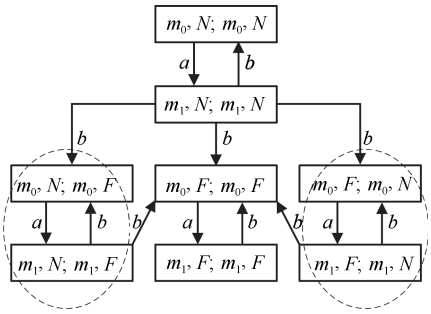

Fig. 4. F-verifier.

VI. Comparison With the Approach in [11]

Cabasino *et al.* propose a general diagnosability test approach in [11] for both bounded and unbounded labeled Petri nets. For the bounded case, they first construct a verifier net (VN) and its reachability graph (RG). Then, cycles in the RG after the occurrence of a fault transition are computed. Note that the number of states in the RG grows exponentially wrt the size of the VN, and the complexity of computing cycles is linear in the sum of the number of states and arcs of the RG.

According to Theorem 2, we know that the complexity of our approach is also linear in the sum of the number of states and arcs of the verifier. In the following, we compare these two approaches from the complexity point of view.

Let $(N, m_0, \mathcal{L})$ be a labeled Petri net satisfying Assumptions A1−A3, n be the number of places in N and $k-1$ be the upper bound of the number of tokens in every place, i.e., $\forall m \in R(N, m_0), \forall p \in P, m(p) \leq k-1$. Then the number of states in the RG and EBRG of N, in the worst case, are k^m and αk^m respectively, where $0 < \alpha \leq 1$. According to the construction of the VN in [11], we know that the upper bound of the number of tokens in every place in VN is still $k-1$. Therefore, the sum of the number of states and arcs of the RG of the VN, and the sum of the number of states and arcs of the verifier are $k^{2m} + k^{4m}$ and $4\alpha^2 k^{2m} + 16\alpha^4 k^{4m}$, respectively. Our approach is preferable to the one proposed in [11] if the following inequality holds:

$$k^{2m} + k^{4m} > 4\alpha^2 k^{2m} + 16\alpha^4 k^{4m}. \quad (1)$$

The solution of (1) is $\alpha < 1/2$. In other words, given a bounded labeled net, our method is preferable to the one proposed in [11] if the number of EBMs is less than half of that of reachable markings.

VII. Conclusion

In this paper, an approach is proposed for testing diagnosability of bounded labeled Petri nets. It reduces the computational complexity of the diagnosability method presented by Cabasino *et al.* [11] from the exponential complexity in terms of the number of nodes of basis reachability diagnoser to the polynomial one. Our future work will focus on extending the proposed approach to deal with large and complex discrete event systems [19]−[23] in a decentralized setting.

References

[1] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Diagnosability of discrete-event systems," *IEEE Trans. Automat. Contr.*, vol. 40, no. 9, pp. 1555−1575, Sep. 1995.

[2] S. B. Jiang, Z. D. Huang, V. Chandra, and R. Kumar, "A polynomial algorithm for testing diagnosability of discrete-event systems," *IEEE Trans. Automat. Contr.*, vol. 46, no. 8, pp. 1318−1321, Aug. 2001.

[3] T.-S. Yoo and S. Lafortune, "Polynomial-time verification of diagnosability of partially observed discrete-event systems," *IEEE Trans. Automat. Contr.*, vol. 47, no. 9, pp. 1491−1495, Sep. 2002.

[4] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. C. Teneketzis, "Failure diagnosis using discrete-event models," *IEEE Trans. Control Syst. Technol.*, vol. 4, no. 2, pp. 105−124, Mar. 1996.

[5] M. P. Cabasino, A. Giua, and C. Seatzu, "Diagnosability of bounded Petri nets," in *Proc. 48th IEEE Conf. Decision and Control, 2009 Held Jointly with the 28th Chinese Control Conf.*, Shanghai, China, 2009, pp. 1254−1260.

[6] G. Jiroveanu and R. K. Boel, "The diagnosability of petri net models using minimal explanations," *IEEE Trans. Automat. Contr.*, vol. 55, no. 7, pp. 1663−1668, Jul. 2010.

[7] T. Ushio, I. Onishi, and K. Okuda, "Fault detection based on Petri net models with faulty behaviors," in *Proc. 1998 IEEE Int. Conf. Systems, Man, and Cybernetics*, San Diego, CA, USA, 1998, pp. 113−118.

[8] S.-L. Chung, "Diagnosing PN-based models with partial observable transitions," *Int. J. Comput. Integr. Manuf.*, vol. 18, no. 2−3, pp. 158−169, Mar. 2005.

[9] Y. L. Wen and M. Jeng, "Diagnosability analysis based on T-invariants of Petri nets," in *Proc. 2005 IEEE Networking, Sensing and Control*, Tucson, AZ, 2005, pp. 371−376.

[10] Y. L. Wen, C. Li, and M. Jeng, "A polynomial algorithm for checking diagnosability of Petri nets," in *Proc. 2005 IEEE Int. Conf. Systems, Man, and Cybernetics*, Waikoloa, HI, 2005, pp. 2542−2547.

[11] M. P. Cabasino, A. Giua, S. Lafortune, and C. Seatzu, "A new approach for diagnosability analysis of Petri nets using verifier nets," *IEEE Trans. Automat. Contr.*, vol. 57, no. 12, pp. 3104−3117, Dec. 2012.

[12] M. P. Cabasino, A. Giua, and C. Seatzu, "Diagnosability of discrete-event systems using labeled Petri nets," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 1, pp. 144−153, Jan. 2014.

[13] T. Murata, "Petri nets: Properties, analysis and applications," *Proc. IEEE*, vol. 77, no. 4, pp. 541−580, Apr. 1989.

[14] M. P. Cabasino, A. Giua, M. Pocci, and C. Seatzu, "Discrete event diagnosis using labeled Petri nets. An application to manufacturing systems," *Control Eng. Pract.*, vol. 19, no. 9, pp. 989−1001, Sep. 2011.

[15] M. P. Cabasino, A. Giua, and C. Seatzu, "Fault detection for discrete event systems using Petri nets with unobservable transitions," *Automatica*, vol. 46, no. 9, pp. 1531−1539, Sep. 2010.

[16] N. Ran, S. G. Wang, H. Y. Su, and C. Y. Wang, "Supervisor synthesis for enforcing linear constraints on a class of Petri nets with unobservable transitions," *IMA J. Math. Control Info.*, 2015. doi: 10.1093/imamci/dnv059

[17] M. P. Cabasino, P. Darondeau, M. P. Fanti, and C. Seatzu, "Model identification and synthesis of discrete-event systems," in *Contemporary Issues in Systems Science and Engineering*, M. C. Zhou, H. X. Li, and M. Weijnen, Eds. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2015, pp. 343−366.

[18] R. Tarjan, "Depth-first search and linear graph algorithms," *SIAM J. Comput.*, vol. 1, no. 2, pp. 146−160, Jun. 1972.

[19] S. G. Wang, D. You, M. C. Zhou, and C. Seatzu, "Characterization of admissible marking sets in Petri nets with uncontrollable transitions," *IEEE Trans. Automat. Contr.*, vol. 61, no. 7, pp. 1953−1958, Jul. 2016.

[20] S. G. Wang, C. Y. Wang, and M. C. Zhou, "Design of optimal monitor-based supervisors for a class of petri nets with uncontrollable transitions," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 43, no. 5, pp. 1248−1255, Sep. 2013.

[21] S. G. Wang, D. You, and C. Y. Wang, "Optimal supervisor synthesis for petri nets with uncontrollable transitions: A bottom-up algorithm," *Inform. Sci.*, vol. 363, pp. 261−273, Oct. 2016.

[22] G. Liu, "Complexity of the deadlock problem for Petri nets modeling resource allocation systems," *Inform. Sci.*, vol. 363, pp. 190−197, 2016.

[23] G. Liu, C. Jiang, and M. Zhou, "Two simple deadlock prevention policies for S3PR based on key-resource/operation-place pairs," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 4, pp. 945−957, 2010.

Ning Ran received the B. S. degree in automation from Hebei University, Baoding, China, in 2010, and the M.S. degree in control theory and control engineering from North China Electric Power University, Baoding, China, in 2013, and is currently pursuing the Ph.D. degree in control science and engineering at Zhejiang University, Hangzhou, China, and is also with the Institute of Cyber-Systems and Control, Zhejiang University. His current research interests include discrete event systems and fault diagnosis.

Hongye Su received the B.S. degree in industrial automation from Nanjing University of Chemical Technology, Nanjing, China, in 1990, and the M.S. and Ph.D. degrees from Zhejiang University, Hangzhou, China, in 1993 and 1995, respectively. From 1995 to 1997, he was a lecturer with the Department of Chemical Engineering, Zhejiang University, where he was an associate professor with the Institute of Advanced Process Control from 1998 to 2000. He is currently a professor with the Institute of Cyber-Systems and Control, Zhejiang University. His current research interests include robust controls, time-delay systems, and advanced process control theory and applications.

Shouguang Wang received B.S. degree in computer science from the Changsha University of Science and Technology, Changsha, China, in 2000, and the Ph.D. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 2005. He joined Zhejiang Gongshang University in 2005, where he is currently a Professor with the School of Information and Electronic Engineering, the director of the Discrete-Event Systems Group and the dean of System Control and modeling Research Institute in Zhejiang Gongshang University. He was a visiting professor with the Department of Electrical and Computer Engineering, New Jersey Institute of technology, Newark, NJ, from Jan. 2011 to Jan. 2012. He is a visiting professor with the Electrical and Electronic Engineering Department, University of Cagliari, Cagliari, Italy, from Dec. 2014 to Dec. 2015. He was the dean of the Department of Measuring & Control Technology and Instrument from July 2011 to July 2014. Corresponding author of this paper.