Congreso Internacional en Ingeniería Electrónica. Mem. Electro 2015, Vol. 37, pp 141-146, Chihuahua, Chih. México
http://depi.itchihuahua.edu.mx/display/memorias_electro/MemoriaElectro2015
ISSN 1405-2172

FAULT DIAGNOSIS IN DISCRETE EVENT SYSTEMS NOT DIAGNOSABLE

K. Hernández-Rueda[1], M.E. Meda-Campaña[2]
Universidad de Guadalajara
[1]CUCSUR-Ingenierías, [2]CUCEA-Sistemas de Información
Av Juaréz 976, Col. Centro
Tel. +523173825010
karenhr@cucsur.udg.mx, emeda@cucea.udg.mx

RESUMEN.

Algunos Sistemas de Eventos Discretos modelados con redes de Petri Interpretadas (RPI) no pueden diagnosticarse si tienen una secuencia de disparo que se puede repetir infinitamente (ciclo indeterminado) después de ocurrida una falta. Por lo que se propone un método para convertir esas redes a una que asegure no tener ese tipo de secuencias de disparo para que pueda ser diagnosticada una falta. La propuesta considera la información de la estructura de la parte repetitiva del sistema para hacer una conversión de una red no diagnosticable a una red diagnosticable. Se presentan ejemplos de algunos sistemas donde las faltas no son diagnosticables y se muestra que cuando se usa el método propuesto se obtiene una RPI diagnosticable.

Palabras Clave: Sistemas de Eventos Discretos, Diagnóstico de faltas, Redes de Petri Interpretadas.

ABSTRACT.

Some Discrete Event Systems that are modeled with Interpreted Petri net (IPN) cannot be diagnosable if have a firing sequence that can be repeated infinitely after a fault occurred (indeterminate cycle). Therefore, a method is proposed to convert this Interpreted Petri net that is not diagnosable to one that can be diagnosable. The proposal consider the structure information of a repetitive part of the system to make the conversion.

Keywords: Discrete Event System, Fault diagnosis, Interpreted Petri nets.

1. INTRODUCTION

Fault diagnosis is an important step in a fault tolerant scheme that allows preserving the system integrity, minimizing risks to humans, and improving the reliability of the system. Even though strict norms and protocols are used to design systems, the absence of faults cannot be guaranteed in any system, thus fault detection and isolation algorithms must be included in systems, in this way the malfunctioning system components can be detected on line and risky system states can be avoided. Nowadays, the Discrete Event Systems (DES) research community has used the model-based approach for fault diagnosis in DES because it does not require detail in-depth the model system to be diagnosed [1]; several works on the matter use finite automata (FA) or Petri nets (PN) as modeling formalism. The use of FA is limited to small size systems [2], [3] and [4], because of the exponential growth of state space (particularly with the concurrent behavior). To cope with the state explosion problem, research groups throughout the world are increasingly adopting PN as a modeling formalism for DES. PN have demonstrated to be one of the most efficient formalism for DES due to its graphic interface (clear graphical description) and its mathematical support for analyzing the model properties like causality, parallelism, synchronization, and analysis concurrency mutexes [4]. Some of the works that use PN for fault diagnosis are the following. In [5],[6] and [7], the information to make the diagnosis is obtained from the inputs and outputs generated by the system. Ramirez in [6] works with the diagnosability of event-detectable live and safe PNs under a structural approach. In [8] is proposed a diagnoser based on the PN paths and causality relationships for determining the presence of faults in a system. In [7], fault monitoring for hybrid PN is addressed; monitoring PN is used to supervise when tasks start, finish, interrupted, are resumed. In [9] it is assumed that there exist not unobservable cycles no blocked firing sequences after the firing of any faulty transition; necessary and sufficient conditions are given for diagnosability based in reachability diagnoser. In [10], fault detection technique based on an identified model is proposed, the online detection and location are done; however, it does not study the diagnosability property. Finally, in [11] a new structural diagnosability characterization is proposed for permanent and operational faults, the new characterization is based on the analysis of siphons leading to procedures that allow livelocks detection, and it is used in this work to diagnose permanent faults. However, these works cannot diagnose nets that have a firing sequence that can be repeated infinitely (indeterminate cycle) after a fault occurred. Therefore, in this work a method is proposed to diagnose systems modeled with Interpreted PN (IPN) that are not diagnosable with the previous theory. The method consist in modify the IPN structure's to one where a firing sequence cannot be fired infinitely after a fault occurred. This approach is based on the analysis of the repetitive part of IPN.

The paper is organized as follows: Section 2 provides basic definitions of DES, PN and IPN. Section 3 presents the theory of diagnosis, diagnosability and modeling of faults. In the section 4 is shown the method that is proposed to diagnose an IPN that is not diagnosable. In the section 5 the results are shown. Finally, in the section 6, the conclusion and future work are presented.

2. BACKGROUND

2.1. Discrete Event System.

A DES is a dynamic system that has a discrete state space and it evolves based on the events occurrence. The state space is countable, however possibly infinite, both time and states are discrete [12]. On the other hand, when the state of a system is naturally described by a discrete set like {0,1,2,..}, and transitions are observed at discrete points in time, it can be associated these state transitions with "events" and then it talks about a "discrete event system". It is shown in figure 1.

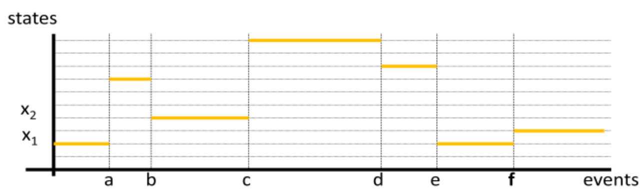

Figure 1. Evolution of state space of a Discrete Event System.

This kind of systems satisfies the following two properties: 1) the state transition mechanism is event-driven (it refers to the fact that the state can only change at discrete points in time, which physically correspond to occurrences of asynchronously generated discrete events) and 2) the state space is a discrete set. This can see in [13]. Examples of these systems are network systems, distributed systems, traffic control systems, manufacturing systems, among others.

2.2. Petri nets.

A PN consists of a network structure (a bipartite digraph), a states description (marking) and a transition rule (the marking game). An example of this kind of net is shown in figure 2. The PN is represented by two kinds of vertices: circles, represent places (p_1,p_2,p_3,p_4), that are associated with actions or system outputs to be modeled and bars o rectangles (t_1,t_2,t_3,t_4) representing transitions, which are associated with events and actions or outputs. Directed arcs connect places to transitions and transitions to places. The places in turn can be marked with black dots into them. An initial marking would be an initial distribution of marks. The presence or absence of a mark in a place can indicate whether a condition associated with this place is true or false. At any given time instance, the distribution of marks into places is called PN marking. The marking defines the current state of the modeled system.

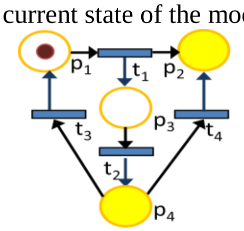

Figure 2. Example of a Petri net.

The input places (output places) are places whose arcs lead to (leave to) a transition t_j and they are considered input (output) of t_j. The PN works in order to simulate the dynamic behavior of a system, the marking in a PN is changed according to the following firing rule: a) a transition "t" is said to be enabled if each input place "p" of "t" is marked with at least w(p,t) tokens, where w(p,t) is the weight of the arc from "p" to "t", b) an enabled transition may or may not fired (depending on whether o not the event related with "t" actually takes place), and c) the firing of an enabled transition "t" removes w(p,t) tokens from each input place "p" of "t", and adds w(t,p) tokens to each output place "p" of "t", where w(p,t) is the weight of the arc from "t" to "p". The formal definition of PN is presented as follows.

Definition 1: A Petri Net structure G is a bipartite digraph represented by the 4-tuple G=(P,T,I,O) where:

- $P = \{p_1, p_2, ..., p_n\}$ and $T = \{t_1, t_2, ..., t_m\}$ are finite sets of vertices called places and transitions, respectively.
- $I(O) : P \times T \rightarrow Z^+$ is a function representing the weighted arcs going from places to transitions (transitions to places); Z^+ is the set of nonnegative integers.

The symbol $\bullet t_j$ denotes the set of all places p_i such that $I(p_i, t_j) \neq 0$ and $t_j\bullet$ the set of all places p_i such that $O(p_i, t_j) \neq 0$. Analogously, $\bullet p_i$ denotes the set of all transitions t_j such that $O(p_i, t_j) \neq 0$ and pi• the set of all transitions t_j such that $I(p_i, t_j) \neq 0$.

The pre-incidence matrix of G is $C^- = [c_{ij}^-]$, where $c_{ij}^- = I(p_i, t_j)$; the post-incidence matrix of G is $C^+ = [c_{ij}^+]$, where $c_{ij}^+ = O(p_i, t_j)$; the incidence matrix of G is $C = C^+ - C^-$. The marking function $M:P^- \rightarrow Z^+$ represents the number of marks (depicted as dots) residing inside each place. The marking of a PN is usually expressed as an n−entry vector. M function can be represented as M(p).

Definition 2: A PN is the pair N =(G, M_0), where G is a PN structure and M_0 is an initial token (mark) distribution over places.

Definition 3: A P−semiflow Y_i (T− semiflow X_i) of a PN is a positive integer solution of the equation $Y_i^T C = 0$ ($CX_i = 0$). The support of the P−semiflow Y_i (T− semiflow X_i) is the set $\|Y_i\| = \{p_j \mid Y_i(p_j) \neq 0\}$ ($\|X_i\| = \{t_j \mid X_i(t_j) \neq 0\}$).

Definition 4: The reachability set of G, denoted by R(G,M_0), is the set of all possible reachable markings from M_0, firing only enabled transitions.

Definition 5: A PN (G, M_0) is *k*-safe (*k*-bounded) if for all $M \in R(G, M_0)$ and places $p \in P$, $M(p) \leq k$. 1 − safe nets are simply called safe.

Definition 6: A PN (G, M_0) is live if for all $M_i \in R(G, M_0)$ and for all $t \in T$ it is true that $\exists\, M_j$, such that $M_i \longrightarrow M_j \longrightarrow$.

Definition 7: A siphon is a subset of places $S = \{p_1, ..., p_s\} \subseteq P$ of a PN such that the set of input transitions $\bullet S$ is contained in the set of output transitions $S\bullet$, i.e., $\bullet S \subset S\bullet$.

2.3. Interpreted Petri net.

An IPN is composed by a PN model together with input and output alphabets; the input and output symbol are system signals assigned to transitions (t_1,t_2) and places $(p_1,\ldots p_4)$, respectively. An IPN can model the commands sequences given by the signals from actuators and sensors each time a new state is reached. Graphically, an IPN is presented in the figure 3. The measurable places (p_1,p_3) of IPN are transparent circles while the no-measurable places $\{p_2, p_4\}$ are dark circles. The formal definition is as follows:

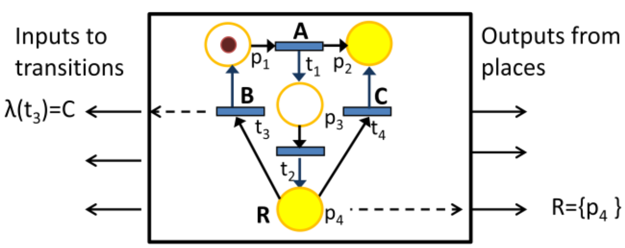

Figure 3. Interpreted Petri net.

Definition 8: An IPN is the 4-tuple $Q = (N, \Sigma, \lambda, \varphi)$ where:

- $N = (G, M_0)$ is a PN.
- $\Sigma = \{\alpha_1, \alpha_2, ..., \alpha_r\}$ is the input alphabet of the net, where α_i is an input symbol.
- $\lambda : T \rightarrow \Sigma \cup \{\varepsilon\}$ is a labeling function of transitions with the following constraint: $\forall t_j, t_k \in T, j \neq k$, if $\forall p_i\ I(p_i, t_j) = I(p_i, t_k) \neq 0$ and both $\lambda(t_j) \neq \varepsilon$, $\lambda(t_k) \neq \varepsilon$, then $\lambda(t_j) \neq \lambda(t_k)$. In this case ε represents an uncontrollable system event.
- There exists a $q \times n$ matrix φ, such that $y_k = \varphi M_k$ is mapping of the marking M_k into the q−dimensional observation vector. Column $\varphi(\bullet, i)$ is the elementary vector e_h if place p_i has associated the sensor place h; or the null vector if p_i has no associated sensor place. In this case, an elementary vector e_h is the q−dimensional vector with all its entries equal to zero, except entry h, that it is equal to 1. A null vector has all its entries equal to zero.

Notice that q places have associated a sensor, signal thus they are measurable or observable.

A transition $t_j \in T$ of an IPN is enabled at marking M_k if $\forall p_i \in P$, $M_k(p_i) > I(p_i, t_j)$. An enabled transition t_j, labeled with a symbol other than ε (empty or silent) symbol, must be fired when $\lambda(t_j)$ is activated. An enabled transition t_j, labelled with a ε symbol can be fired. When an enabled transition t_j is fired in a marking M_k, then a new marking M_{k+1} is reached. This fact is represented as $M_k \xrightarrow{t_j} M_{k+1}$; M_{k+1} can be computed using the dynamic part of the state equation represented by (1):

$$\begin{aligned} M_{k+1} &= M_k + Cv_k \\ y_k &= \phi M_k \end{aligned} \qquad (1)$$

Definition 9: A firing transition sequence of an IPN (Q,M_0) is a sequence $\sigma = t_i t_j \ldots$ such that $M_0 \xrightarrow{t_i} M_1 \xrightarrow{t_j} \ldots$ The set of all firing sequence $£(Q,M_0)$, is called the firing language of (Q,M_0). $£(Q,M_0) = \{\sigma \mid \sigma = t_i t_j \ldots \text{ where } M_0 \xrightarrow{t_i} M_1 \xrightarrow{t_j} \ldots\}$.

Definition 10: A sequence of observation vectors (output symbols) of (Q,M_0) is a sequence $\omega = (y_0)\,(y_1)\,\ldots\,(y_n)$, where $y_k = \phi\, M_k$ and $y_i \neq y_{i+1}$. If ω is a sequence of output symbols, then the set of firing transition sequences $\sigma \in £(Q,M_0)$ whose firing generates the output sequence ω is represented by $\Omega(\omega)$.

Definition 11: Let (Q,M_0) be an IPN. The set $\Lambda(Q, M_0)$ denotes all sequences of output symbols of (Q,M_0). The set of all output sequences of length greater than or equal to k will be denoted by $\Lambda^k(Q, M_0)$, i.e., $\Lambda^k(Q, M_0) = \{\omega \in \Lambda(Q, M_0) \mid |\omega| \geq k\}$.

Definition 12: The set of all output sequences leading to an ending marking in the IPN (Q, M_0) is denoted by $\Lambda_B(Q, M_0)$, i.e., $\Lambda_B(Q, M_0) = \{\omega \in \Lambda(Q, M_0) \mid \exists \sigma \in \Omega(\omega)$ such that $M_0 \xrightarrow{\omega} M_i$ and M_i enables no transition, or when M_i enables t_i $(M_0 \xrightarrow{t_i})$ then $C(\bullet, t_i) = \vec{0}\}$.

3. DIAGNOSIS AND DIAGNOSABILITY.

A fault in a system is the occurrence of an event that is not registered as a normal system functionality, which firing does not reduce the system performance. Although, faults do not deviate the system from its requirements, the faults must be detected, located and isolated, since their firing could lead the system into an error. The fault can be detected as in figure 4.

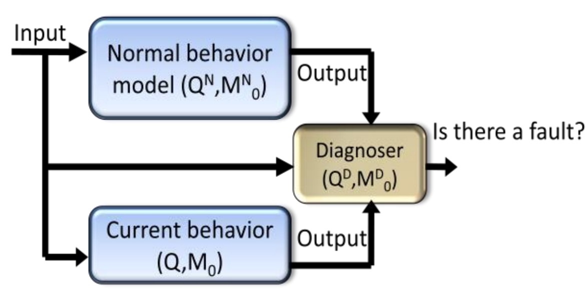

Figure 4. Online diagnosis based on model.

The diagnosis system has a diagnoser (a monitoring online system whose warns the presence of faults) that compares the normal behavior (without faults) modeled with IPN with the current behavior (with possible faults) of a DES, when there exists a difference between these behaviors, then a fault is detected and it can be seen as an error in the DES. Before to make a diagnoser, it is necessary to determine if the DES is event-detectable and diagnosable [11]. The event-detectable property allows to distinguish different events into the system to diagnose faults, i.e. that a DES does not have a firing sequence that can be fired infinitely. Furthermore, if the information of the input and output from the DES it is sufficient to determine whether or not a fault is presents in the DES then the DES is considered diagnosable. This processes is called diagnosability and the test to determine if the system is diagnosable is known as the diagnosability property.

3.1. Diagnosability and event-detectable properties.

The input-output diagnosability property is defined as follows [6]:

Definition 13: An IPN given by (Q, M_0) is said to be input-output diagnosable in $k < \infty$ steps if any faulty marking M_f is distinguishable from any other $M_k \in R(Q, M_0)$ using $\Lambda^k(Q, M_0) \cup \Lambda_B(Q, M_0)$.

This definition is the same than that presented in [3] from the IPN point of view. In fact if an indeterminate cycle appears in the reachability graph or an undetermined blocking marking (blocking markings with more than one faulty label or faulty and normal labels) appears, then the IPN is not input-output diagnosable. The proposal of this paper is devoted to diagnose IPN with an indeterminate cycle. If the IPN that modeled the system is not event-detectable then it is possible that has a firing sequence can be fired infinitely. The event-detectable property is defined as follows [14]:

Definition 14: An IPN (Q, M_0) is event-detectable iff $\forall\sigma \in £(Q,M_0)$, the firing of any pair of transition $t_i,t_j \in \sigma$, can be distinguished from each other using the information in $\omega \in \Lambda(Q, M_0)$.

The following lemma gives a polynomial characterization of event-detectable IPN that is used in order to determine if an IPN is diagnosable [6]. This means that if an IPN is not event-detectable it is impossible to know if it can be diagnosable.

Lemma 1: A live IPN given by (Q, M_0) is event detectable iff

- $\forall t_i, t_j \in T$ such that $\lambda(t_i) = \lambda(t_j)$ or $\lambda(t_i) = \varepsilon$ it holds that $\varphi C(\bullet, t_i) \neq \varphi C(\bullet, t_j)$, and
- $\forall t_k \in T$ it holds that $\varphi C(\bullet, t_i) \neq 0$.

3.2. Modeled faults

The events to be diagnosed are referred to as “faults”, hereafter are modeled as unobservable events in the respective system modules. Events are unobservable when they are not directly recorded by the sensors attached to the system. The objective is to diagnose the occurrence of fault events based on the sequence of observed events. Furthermore, the faults that are considered here are permanent ones.

Definition 15: A permanent fault occurs when a task stops its execution while other(s) taks(s) can be continue to run in the system.

This work deals with systems whose normal behavior model (Q^N, M_0^N) can be represented by a live and safe IPN. The liveness cannot be tested efficiently in IPN (or PN); however, if the modeling methodology proposed in [6] is applied and the conditions indicated on [11] are preserved in the IPN circuits, then the generated (Q^N, M_0^N) is a live and safe IPN.

Once the DES is described by a live and safe IPN, the next step is to represent faults into the normal behavior model. This is done based on [11]. When a permanent fault occurs, then one task is stopped while other concurrent task may continue their execution. In permanent faults, the involved faulty devices will remain in a faulty state until they are repaired. The proposed modeling strategy for representing faults is straightforward. Consider first the model (Q^N, M_0^N) that describes the normal functioning of the system. Then for every place p_i^N representing an operation at which a fault may occur, add an uncontrollable transition t_f, a faulty place p_i^F, and the arcs (p_i^N, t_f) and (t_f, p_i^F). The new faulty place p_i^F must be labeled with the same symbol that p_i^N for stating that the fault cannot be detected from the observation of the outputs. The obtained model describes both normal and faulty behavior as it can see in figure 5b.

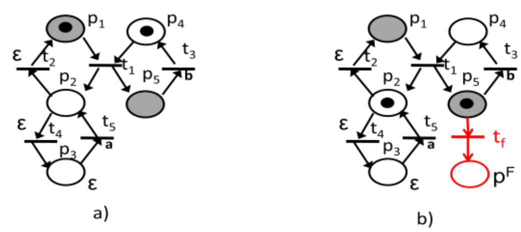

Figure 5. An IPN a) with normal behavior and b) with normal and faulty behavior.

The set of places P of an IPN (Q, M_0) is partitioned into two subsets, $P = P^F \cup P^N$ where P^F is the set of places coding faulty states, and P^N is the set of places coding normal states of the IPN. The markings in $R(Q, M_0)$ can also be partitioned into the following two subsets: $F = \{M \in R(Q, M_0) \mid \exists p_k \in P^F$ such that $M(p_k) > 0, M \in R(Q, M_0)\}$ and $R(Q^N, M_0^N) = R(Q, M_0) - F$, where F is the set of the faulty markings and $R(Q^N, M_0^N)$ is the set of the normal states. The embedded normal behavior IPN (Q^N, M_0^N) of (Q, M_0) is the IPN included in (Q, M_0) when P^F and $T^F = \bullet P^F$ are not considered. In (Q^N, M_0^N) the set of places is $P^N = P - P^F$, the set of transitions is $T^N = T - T^F$ and the set of arcs of (Q^N, M_0^N) is $A^N = ((P^N \times T^N) \cup (T^N \times P^N)) \cap (A)$, where $A = \{(p_i, t_j)|p_i \in P, t_j \in T \text{ and } I(p_i, t_j) = 1\} \cup \{(t_i, p_j)|p_i \in P, t_j \in T \text{ and } O(p_i, t_j) = 1\}$.

Definition 16: Let (Q, M_0) be an IPN, P^N be the normal set of places, and T^F be the set of faulty transitions of (Q, M_0). The set of risky places of (Q, M_0) is $P^R = \bullet T^F$. The post-risk transition set of (Q, M_0) is $T^R = \{P^R\bullet \cap T^N\}$.

Considering last definitions, the next proposition [11] is used in order to determine if a permanent fault can be diagnosable.

Proposition 1: Let (Q, M_0) be a safe (Q^N, M_0^N) that is safe, live and strongly-connected. Let t_i be a permanent fault, p_k be a risky place and S_{ti} be the siphon that will be unmarked when t_i is fired. Assume that $| p_k\bullet| = 1$ and the post-risky transition $t_a \in p_k\bullet$ and the pre-risky transitions are event detectable. (Q, M_0) is diagnosable with respect to t_i if all the T-semiflow of the net contains transitions in $\bullet S_{ti} \cup S_{ti}\bullet$.

3.3. DES not diagnosable

Consider the figure 5b, a live, safe and strongly connected IPN. After the transition t_1 is fired, the initial marks in p_1 and p_4 are moved to the places p_5 and p_2, so if a permanent fault occurs (t_f is activated) a mark will be in the place p^F (this mark is lost) and will not be possible to detect it, because the mark in the place p_2 can fire the transition t_4 and the mark in the place p_3 can fire the transition t_5. This means, the firing sequence transition $\sigma=t_4t_5$ can be fired infinitely (indeterminate cycle) and it is impossible to detect the fault. This IPN does not satisfy the *proposition 1*, where $t_i=t_f$, $p_k=p_5$, $t_3=t_a$, $| p_5\bullet|=1$, and t_3 and t_1 are event-detectable. However, the T-semiflows $X_1=\{t_1,t_2,t_3\}$ and $X_2=\{t_4,t_5\}$ have transitions t_1,t_2 and t_3 that there are not related with $S_{ti}=\{p_2,p_3\}$ then not all transitions are in $\bullet S_{ti} \cup S_{ti}\bullet$. So this net is not input-output diagnosable with respect to t_F.

4. PROPOSAL

In this section is presented the method that is proposed to diagnose faults in IPN with an indeterminate cycle.

The idea in order to eliminate the indeterminate cycle to diagnose this net is the following:

1. Since (Q^N, M_0^N) is live, safe and strongly connected IPN, then all places belong to a P-semiflow.
2. When a place is added, a new P-semiflow is formed and it is possible to create a new T-semiflow that is the result of the existing T-semiflows (the new T-semiflow is a sum of the previous T-semiflows).
3. If the new IPN preserve the properties of liveness, safety and strongly connected, then it is possible to apply the *proposition 1* and determine if this new net is diagnosable.
4. When a fault transition is added to (Q^N, M_0^N), then some P-semiflows are transformed into siphon. Thus the siphon containing a risky place p_k can be computed as a P-semiflow in (Q^N, M_0^N).
5. So the transitions related to the siphons can be easily computed as the inputs to the P-semiflows. Afterward when there exists a T-semiflow sharing transitions with the transitions of the siphon, then the fault included in the siphon is diagnosable.

4.1. Method proposed.

It is presented the next method to eliminate the indeterminate cycle that appear in figure 5b, formed by the transitions $\{t_4,t_5\}$ and the places $\{p_2,p_3\}$. This is based on the *definition 16* and the *proposition 1*.

Method 1: convert an IPN not diagnosable to one diagnosable.

1. Obtain a safe and live IPN model of a DES that has an indeterminate cycle.
2. Calculate the P-semiflows and T-semiflows of the IPN.
3. Identify P-semiflows and T-semiflows that are part of the indeterminate cycle, the places and transitions.
4. Add permanent faults. Notice that the places of (Q^N, M_0^N) connecting faulty transitions belongs to siphons.
5. Identify the risk place p_k.
6. Review that the post-risky transition $t_a \in p_k\bullet$ and the pre-risky transitions are event detectable.
7. Add two new places that must be connected between the post-risky transition and a transition of the indeterminate cycle. The transition of the indeterminate cycle does not belong to the T-semiflows existing or does not belong more than one T-semiflow. Each place must be independent, this means, each one must have, an arc from (to) the transition of the indeterminate cycle to (from) that place whose must be connected to (from) the post-risky transition.
8. The new places are added to C matrix and must satisfy the equation $C[X_1+X_2... +X_i]=0$ in order to obtain the new T-semiflow $X_r= X_1+X_2... +X_i$. Where $X_1+X_2... +X_i$ is the number of T-semiflows of the IPN original and X_r is the new T-semiflow.
9. Each new place must have a token.
10. Eliminate the permanent faults in order to simulate the new IPN.

4.2. Cases

Consider the next IPNs from figure 6. These nets are asymmetric choice and have an indeterminate cycle; formed by (p_5,p_6) to the IPN1, by (p_3,p_4) to the IPN2 and by (p_2,p_4) to the IPN3. The permanent fault t_F is not diagnosable in these nets. They were simulated in pntool of Matlab to identify some characteristics like a class of the net, P-semiflows and T-semiflows before use the method. The places with marks (represented with number 1) are the initial marks. The P-semiflows and T-semiflows are represented by Y_1,Y_2 and by X_1,X_2 respectively.

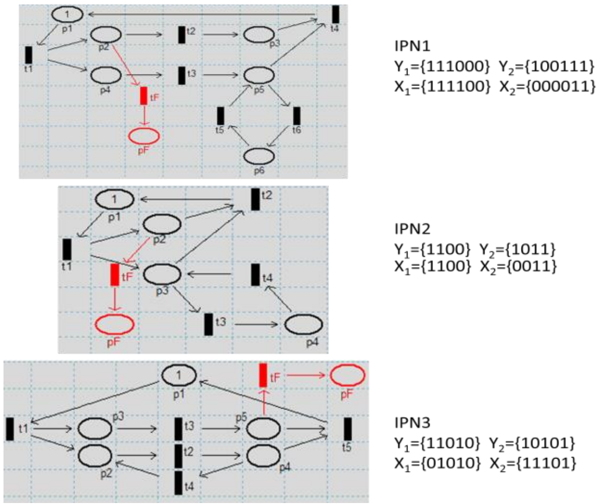

Figure 6. Some IPN that have an indeterminate cycle.

5. RESULTS

After the *method 1* is used, the new nets obtained are depicted in figure 7.

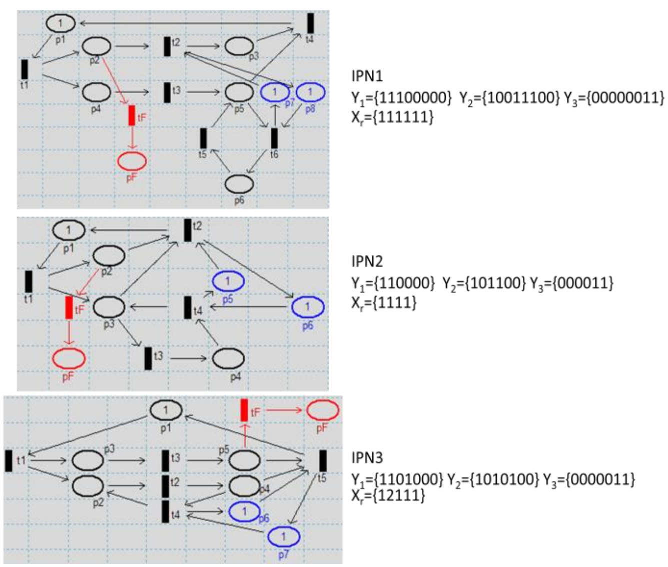

Figure 7. New IPNs from figure 6, formed with the removed indeterminate cycle.

The indeterminate cycle is removed adding new two places; (p_7,p_8) by the IPN1, (p_5,p_6) by the IPN2 and (p_6,p_7) by the IPN3. Therefore, the permanent fault t_F can be diagnosable. It is obtained an additional P-semiflow formed by the two new places and one T-semiflow Xr obtained as the sum of T-semiflows from the original IPN. In these nets the properties of live, safe and strongly connected are remain.

6. CONCLUSIONS

A method to convert a DES not diagnosable to a DES diagnosable has been proposed. It based on the analysis of T-semiflows of the IPN that represents the DES in order to change its structure, i.e. eliminate the indeterminate cycle that can be fired after a fault has occurred. This elimination is done adding two new places to the IPN. These places are connected using the post-risk transition and one transition that belong to the indeterminate cycle. As it can see in the results, the IPNs asynchronous choice with an indeterminate cycle can convert to one without it. Also, these nets are diagnosable. In future work is considered reviewing others IPNs in order to generalize the results and determine the conditions that are needed to have a IPN diagnosable.

REFERENCES.

[1] S. Lafortune, D. Teneketzis and M. Sampath, "Failure Diagnosis of Dynamic Systems: An approach based on Discrete Event Systems", Proceedings of the American Control Conference, 2001 pp. 2058-2068.

[2] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen and D. Teneketzis, "Diagnosability of discrete event systems". IEEE Transactions on Automatic and Control, vol 4, no. 9, 1995, pp. 1555-1575.

[3] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen and D. Teneketzis, "Failure Diagnosis Using Discrete-Event Models", IEEE Transactions on Control Systems Technology. Vol. 4, No. 2, 1996, pp.105-124.

[4] R. Debouck,"Diagnosis of Discrete Event System. A modular Approach". Proceedings of the IEEE Conference on System, Man and Cybernetics, 2003, pp. 306-3011.

[5] L. C. Chung and M. Jeng, "Failure Diagnosis: A case study on Modeling and Analysis by Petri nets". Proceedings of IEEE Conference on Systems, Man and Cybernetics, 2003, pp. 2727-2732.

[6] A. Ramírez-Treviño, E. Ruiz-Beltrán, I. Rivera-Rangel and E. López-Mellado, "Online Fault Diagnosis of Discrete Event System. A Petri Net Based Approach". Transaction on Automation Science and Engineering Vol. 4, no. 1, 2007, pp. 31-39.

[7] M. Dotoli, M.P. Fanti, A.M. Mangini and W. Ukovich, "On-line Fault Detection in DES by Petri nets and Integer Linear Programming". Automatica. Vol. 45. no.11, 2009, pp. 2665-2672.

[8] Lefebvre D. and Delherm C. "Diagnosis of DES with Petri net models". IEEE Transaction on Automation Science and Engineering Vol. 4, no. 1, 2007, pp. 114-118.

[9] Cabasino, M., Giua, A., Lafortune, S., and Seatzu, C. "Diagnosability analysis of unbounded petri nets". Proceeding of the 48th IEEE Conference in Decision and Control 2009, pp. 1267-1272. doi:10.1109/CDC.2009.5400608.

[10] D. Lefebvre and E. Leclerq, "Stochastic Petri net identification for the fault detection and isolation of discrete event systems". IEEE Transaction on Systems, MAN, Cybernetics, A., Syst. Humans, Vol. 41, no. 2, 2001, pp. 213-225.

[11] E. Ruiz-Beltrán, A. Ramirez-Treviño and J.L. Orozco-Mora. "Fault Diagnosis in Petri Nets". Formal Method in Manufacturing. CRC Press Taylor-Fracis Group, Boca Raton, FL. 2014. pages 728.

[12] M. Silva, Las redes de Petri: en la automática y la informática. Editorial AC, Madrid, España, 1985, pp. 558-562.

[13] C.G. Cassandras and S. Lafortune, Introduction to Discrete Event Systems, Editorial Springer. Second edition. New York, USA, 2008.

[14] Ramírez-Treviño A., Ruiz-Beltr_an E., Ar_amburo J. and López-Mellado E. Structural Diagnosability of DES and Design of Reduced Petri Net Diagnosers. IEEE Transaction on Systems, MAN and Cybernetics. Vol. 42, No. 2, 2012, pp. 416-429.