Fault Diagnosis Using Petri Nets. A case study

Karen Hernández Rueda
University of Guadalajara, Zapopan, Jalisco, México, karenhr@cucsur.udg.mx

María Elena Meda Campaña
University of Guadalajara, Zapopan, Jalisco, México, emeda@cucea.udg.mx

ABSTRACT

The motivation of this work is to ensure that a Discrete Event System continues working after a fault occurs, since some systems, like power plants, should not stop working because they can generate service interruptions that cause severe economic impacts and even dangers. So, the interest of this paper is to analyze the Discrete Event System model to detect faults using Interpreted Petri nets and here is presents a case study about how can be uses a diagnosis scheme in order to identify if there exists a fault, considering permanent and control faults that are modeled with Petri nets.

Keywords: Fault diagnosis, diagnosis scheme, fault detection, Petri net modeling

1. INTRODUCTION

This paper addresses the fault diagnosis problem in the behavior of a Discrete Event System (DES) that is modeled with Interpreted Petri nets (IPN). The fault of a system component does not directly lead to system failure, but may be the beginning of a serie of failures that maybe they end up with system failure.

The need for accurate and timely diagnosis of DES failures, in the interests of safety, reliability, and economy, has prompted widespread interest in the area of failure diagnosis both in industry and in academia. A great deal of research effort has been and is being spent on the design and development of automated diagnostic systems. A variety of schemes, differing both in their theoretical framework and in their design and implementation philosophy, have been proposed. From the conceptual viewpoint most existing methods of failure diagnosis can be classified as: 1) fault-tree based methods; 2) quantitative, analytical model-based methods; 3) expert systems; 4) model-based reasoning methods; and 5) methods based on system model discrete events (Sampath et al, 1998).

The advantage of the system modeling approach is that it no requires detail in-depth the model system to be diagnosed and it is ideal for diagnosis of systems that are difficult to model. The problem of fault diagnosis for DES using diagnosis methods based on discrete event models have been successfully used in a variety of technological systems (Lafortune et al, 2001). The "diagnosis approach", introduced in (Sampath et al, 1996) and extended in several works like in (Ruiz et al, 2007) it is important for this paper because it takes its bases to analyze the fault diagnosis problem. The key feature of the diagnosis approach is the use of a special discrete event process named "the diagnoser". The diagnoser is built from the system model and is used to 1) test the diagnosability property of the system and 2) perform online monitoring of the system for the propose of fault diagnosis (Genc and Lafoturne, 2007).

The design process and/or analysis of DES is done with the generation of a IPN model to verify that the system meets the specifications needed or if has the desired properties such as liveness, safety, reachability among others (Murata, 1989). The interest of this paper is to analyze the DES model to detect faults using IPN (Xiaoli1 et al 2009), so this paper shows how to face up this problem. In particular, it is presented a case study about how can be uses a diagnoser model based on the work presented in (Ruiz, 2007) in order to detect a fault, considering the

properties of diagnosability and detectable-event. First, it is presented the background, later is introduced concepts of Petri nets and fault diagnosis problem, one section is devoted to the case study to explain how to detect faults and finally, a conclusion is presented.

2. BACKGROUND

DES is a system whose state changes abruptly with the occurrence of events. DES encompass a wide variety of systems such as manufacturing systems, transportation systems, supply chain networks, operating systems, and communication systems, between others (Silva 1985). These systems are dynamic and change their states with the occurrence of discrete events. All these systems are designed with controllers to ensure its normal behavior. DES controllers are used to restrict the behavior of DES to a desirable set of behaviors that do not violate the DES control specifications or constraints. However, they can fail.

There are different tools used to model the structure and the dynamics of the DES, as Petri nets (PN) and finite automaton (FA). Petri nets are considered a formal tool suitable for carrying out the study of fault diagnosis in DES. It is widely used by the community of computer science (Peterson, 1981) because they describe the behavior of DES, thanks to its graphical nature and mathematical support, as the capture of the characteristics of causality, parallelism, synchronization, and concurrency mutexes (Debouk, 2003). Some of the most important works that use PN to analyze DES, are mentioned at the following lines.

T. Ushio, I. Onishi and K. Okuda in (Ushio et al, 1998), adopt for the first time the diagnosis approach proposed in (Sampath et al, 1995-1996) and move to the formalism of PN. They represent the normal behavior and failure of a system with PN. The failure behavior is forced by internal actions of the system. Then in (Chung et al, 2003), (Chung et al, 2005) proposed an extension of the work presented by Ushio. Information to make the diagnosis is obtained from the inputs and outputs generated by the system. The disadvantage of the previous works is that it has to perform reachability analysis of diagnosis (finite automaton) obtained from the PN model, and then determine the diagnosability of the PN as in (Sampath et al, 1995).

As mentioned in (Ruiz, 2007), other important works are proposed by C. N. Hadjicostis (Hadjicostis et al, 2002), which proposes a method for fault tolerant systems based on the construction of redundant systems (they are modeled with PN and the original system). The model is used for redundant error correction and robust performance in the system, despite the flaws. However, only the monitoring is done into the systems that are fully measurable and do not analyze the diagnosability property of a system.

In (Ramirez et al, 2004) and (Ramirez et al, 2007) fault diagnosis is performed using the DES model by IPN, which represents the normal and abnormal behavior (lack) of the system. Propose a diagnosis that ensures detect and locate faults in a finite number of steps using a structural characterization for troubleshooting. In (Ruiz et al, 2004) is proposed a bottom-up methodology ("bottomup") which, diagnosable models that are built with submodels of themselves that are diagnosable. The methodology ensures that the global model obtained by the rules of composition is diagnosable, which avoids carrying out the analysis of diagnosability of the global model. In (Ruiz et al, 2005) is proposed a new scheme, which allows the development of a diagnosis capable of carrying out the detection and localization of faulty line. The advantages of this work, over previous work, is that carried out the structural analysis model to determine the diagnosability of a DES modeled with IPN and offers an online diagnosis scheme that are easy to implement.

3. PETRI NETS AND FAULT DIAGNOSIS CONCEPTS

Petri nets (PN), as graphical and mathematical tools, provide a uniform environment for modeling, formal analysis and design of DES (Silva, 1985). One of the major advantages of using Petri net models is that the same model is used for the analysis of behavioral properties and performance evaluation, as well as for systematic construction of discrete event simulators and controllers.

A PN may be identified as a particular kind of bipartite directed graph populated by three types of objects. These objects are places (p_1,p_2,p_3,p_4,p_5), transitions (t_1,t_2,t_3,t_4) and directed arcs connecting places to transitions and

transitions to places. Arcs are labeled with their weights (positive integers). Labels for unity weight are usually omitted. Also, there exist marks (tokens) in some place(s) that are represented with black points. See figure 1.

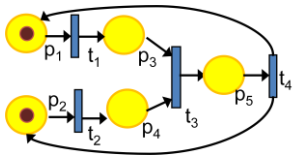

Figure 1: A Petri net

A place is an input place to a transition, if there is a directed arc connecting this place to the transition. A place is an output place of a transition, if there is a directed arc connecting the transition to the place. For instance, input (output) places may represent pre-conditions (post-conditions), and the transition an event. Input places may represent the availability of resources, the transition their utilization, output places the release of the resources. The marking in a PN is changed according to the following firing rule: 1) 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". 2) an enabled transition may or may not fire (depending on whether o not the event actually takes place) and 3) 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 is also used and is presented as follows.

Definition 3.1a A PN structure is a 4-tuple N=(P,T,I,O) where:

- P= $\{p_1,p_2,\ldots p_n\}$ is a finite set of n places
- T= $\{t_1,t_2,\ldots t_m\}$ is a finite set of m transitions
- I: P X T$\rightarrow$ $\{0,1\}$ is a function that represents the arcs of the places to the transitions, and
- O: T X P $\rightarrow$ $\{0,1\}$ is a function that represents the arcs of the transitions to places.

Definition 3.1b The function marking M:P$\rightarrow$ N^+ U $\{0\}$ represents the number of marks (represented like points) inside each place. The marking of a PN represents the state of the system.

Definition 3.1c A PN is the pair **(N,Mo),** where N is the structure of the PN and Mo is the distribution of initial marking.

Definition 3.1d A firing sequence of (N, Mo), is a sequence of transitions $\sigma=t_it_j\ldots t_k$ such that $Mo\xrightarrow{t_i}M_1\xrightarrow{t_j}\ldots M_w\xrightarrow{t_k}\ldots$

Definition 3.1e A Parikh vector is $\vec{\sigma}$: $T\rightarrow (Z^+)^m$, where m=|T|, considering that $\sigma = t_it_jt_k\ldots$ is a firing sequence, $\vec{\sigma}$ maps each transition t Є T in the occurrences number of t in σ.

PN as mathematical tool possess a number of properties. These properties, when are interpreted in the context of the modeling systems, allows the system designer, identify the presence or absence of specific functional properties of the system, under design. Two types of properties can be distinguished: behavioral and structural properties. The behavioral properties are those which depend on the initial state, or marking, of a PN. The structural properties, on the other hand, depend on the topology, or net structure, of a PN. In the following lines is provided an overview of some of the most important (from practical point of view), behavioral properties.

3.1 SOME DYNAMICS PROPERTIES

The following properties ensure that a transition sequence can be fired in the PN and these properties (liveness, strongly-related, limited) are considered into IPN, when the diagnosability and event-detectable properties are defined.

Definition 3.1.1a A PN (N,Mo) is cyclic if $\forall M_i$ Є R(N,Mo) it is true that Ǝ σ, such that $M_i\xrightarrow{\sigma}M_o$.
Definition 3.1.1b A PN (N,Mo) is live if $\forall M_i$ Є R(N,Mo) and $\forall t$ Є T it is true that Ǝ M_j, such that $M_i\xrightarrow{\sigma}M_j\xrightarrow{t}$.

Definition 3.1.1c A PN (N,Mo) is k-safe (k-bounded) if $\forall$ M Є R(N,Mo) and $\forall$ p Є P, M(p) ≤ k. If it is true that $\forall$ M Є R(N,Mo) and $\forall$ p Є P, M(p) ≤ 1, then the net is called 1-safe (safe or binary).

Definition 3.1.1d A PN(N,Mo) is strongly-related if and only if from any marking M_j Є R(N,Mo) it can be reached any other marking M_k Є R(N,Mo) with the firing of a sequence σ_k, i.e. $M_j \xrightarrow{\sigma_k} M_k$.

3.2 INTERPRETED PETRI NETS

In the case study that is presented here, it is used the Interpreted Petri Net (IPN), an extension to the PN that induces the input and output signals of a DES. The IPN is composed by a PN model together with input and output alphabets; these are assigned to transitions and places, respectively. An IPN can model the commands sequences given by the signals from actuators and sensors each time a new state is reached.

Definition 3.2a An Interpreted Petri Net is the 4-tuple Q= (G,Σ,Φ, λ,φ) where,

- G=(G,Mo) is a structure of PN and Mo is the initial marking.
- $\Sigma = \{\alpha_1, \alpha_2, \ldots \alpha_r\}$ is an alphabet of the input symbols, where α_i is the i-th symbol of the input alphabet.
- $\Phi = \{\Phi_1, \Phi_2, \ldots \Phi_v\}$ is an output alphabet.
- λ: T→ Σ U {ε} is a labeling transition function with the restriction: $\forall$ t_j, t_k Є T, j≠k T 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 a null event.
- φ:R(N,M0) → $\{\Phi \cup \{\varepsilon\}\}^q$ is an output function. Э a matrix φ of q x n dimensions, such that $y_k = \varphi M_k$ is the map of marking M_k in a observation q-dimensional vector. The column φ(•,i) is the elemental vector e_h if the place p_i has associated the sensor h; or the null vector if p_i has no associated a sensor. In this case an elemental vector eh is the q-dimensional vector with all entries equal to zero, except the entry h, which is equal to 1. A null vector has all entries equal to 0.

Definition 3.2b A transition t_j Є T of a PN is enabled in the marking M_k if $\forall$ p_i Є P, $M_k(p_i) \geq I(p_i, t_j)$. If $\lambda(t_j) = \alpha_i \neq \varepsilon$ is present and t_j is enabled, then t_j can be fired. If $\lambda(t_j) = \varepsilon$ and t_j is enabled then t_j 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 means: $M \xrightarrow{t_j} M_{k+1}$ y M_{k+1} can be calculated through the state equation of an IPN as:

$$M_{k+1} = M_k + C\vec{v_k} \quad \text{and} \quad y_k = \varphi(M_k)$$

Where C is the incidence matrix and $\vec{v_k}$ is a firing vector, which was defined as PN previously. Therefore, y_k Є $(Z^+)^q$ is the k-th observation vector of a PN.

Definition 3.2c If a $\lambda(t_i) \neq \varepsilon$ the transition t_i is said manipulated; in other case t_i is no manipulated. **A place p_i Є P is measurable if the i-th column of the column vector of φ is no null**, i. e. $\varphi(\bullet, i) \neq \vec{0}$;otherwise it is no measurable.

3.3 DIAGNOSABILITY

The diagnosability concept is introduced in (Sampath, 1995) with the definition of the fault diagnosis problem using FA. The property of diagnosability of DES is linked to the ability to detect the occurrence of certain events different, non-observable (fault events), from sequences of observable events. The diagnosability can be verified with the implementation of a diagnoser. As was mentioned in the introduction, the diagnoser is built from the system model and is used to test the diagnosability property of the system and perform online monitoring of the system for the purpose of fault diagnosis. The diagnosability of a system is evaluated before the building the diagnoser model and is necessary to consider when a system can be diagnosable. In this case, it is used the theorem from (Ruiz, 2007) and this is defined as follows.

Theorem: Let (Q,Mo) be a safe IPN with the permanent, intermittent and control faults, where (Q^N, M^N_0) together with the control faults transitions is a safe, live and event-detectable IPN. If

- $\forall$ t_i Є T^R, $\forall$ t_j Є T^R where $t_i \neq t_j$, the maxima relative distance D_H between these transitions are finite.
- $\forall$ t_k Є T^R , $\bullet(t_k) = \{p^N_i\}$, it must fulfill that $|\bullet(t_k)| = 1$ and $\lambda(t_k) \neq \varepsilon$.

Then (Q, Mo) is input-output diagnosable.

Also, it is necessary to consider the following definition (Ruiz, 2007):

Definition: Let (Q,Mo) be an IPN, P^N is the places set of normal behavior and T^{PF} is the transitions set of permanent failures of (Q,Mo). The places set of risk of (Q,Mo) is the set $P^R={}^\bullet T^{PF}$. The transitions set post-risk of (Q,Mo) is the set $T^R=\{P^{R\bullet} \cap T^N\}$. Where T^N is the normal transitions set $T^N=T-(T^{PF} U T^{CF})$ and $P^N=P-P^{PF}$, P^{PF} is the places set that distinguishes places of permanent failures.

Where the D_H is the maximum distance between two transitions (t_i and t_j) in the two directions, this mean, the number of firing t_i when the mark is retained in the $\bullet t_j$ (input place of the transition t_j) and vice versa. This theorem means that two transitions, that can be the transitions of post-risk must have only one input place and they must not be labeled with the ε- label, and the maximum relative distance between them are finite (there no exist indeterminate cycles). In order to use this theorem, it is necessary to consider the following lemma (Rivera et al, 2007).

Lemma: A live IPN (Q,Mo) is event-detectable if and only if

- $\forall t_i, t_j \in T$ 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 $\varphi C(\bullet, t_k) \neq 0$.

This means, there is no possible that two column vectors can be equals and furthermore, all the columns should be different from zero.

3.3.1 DIAGNOSIS SCHEME

The diagnosis approach as a system modeling approach needs a diagnosis squeme. This considers the system model in the normal behavior and the diagnoser (Ramírez et al, 2007). It can be observed in the figure 2.

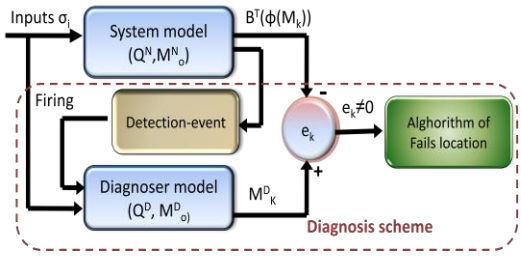

Figure 2: Diagnosis scheme

When certain non-manipulable and manipulable inputs are given in the system, these inputs generate a marking change in the system model. These inputs will affect the output of the system model, which contains the normal behavior and failure of the system, and also they will affect the output of the diagnoser model that contains only the good behavior of the system. In case any fault occurs in the system, then the error (e_k) between the output of the system model and diagnoser model will be zero ($e_k=0$), indicating that no fault is present in the system. The existence of a system failure is detected when the error $e_k \neq 0$, so it is necessary to calculate the error e_k to determine whether there was a failure. Considering the work of (Ramirez et al 2007), once that it has the model of normal system behavior (Q^N,M^N_0), the diagnoser (Q^D, M^D_0), the initial marking of the diagnoser ($M^D_0= B^T \varphi Mo$ where B^T is a vector of nonnegative entries of qx1) and firing rules of transitions diagnosed, then it can calculate the error. The firing rules of transitions are; if a transition $t_j \in T^R$ is enabled on (Q^D, Mo^D) and $\lambda(t_j)$ is turned on (Q,Mo), then t_j must be fired in (Q^D,M^D_0). Thus, if t_j does not fire on (Q, Mo), then there exist an error in (Q,Mo) and the observation vector of the system model and the observation vector of the diagnoser model will be different.

The error is calculated considering the state equation of the network containing the measurable part of the system model with the model-diagnosis as: $[M^D_k, \varphi M_k]^T = [M^D_0, \varphi M_0]^T + [C^D, \varphi C]^T v_k$ *(eq.1)*. $C^D= B^T \varphi^N C^N$ is an incidence matrix of (Q^D, M^D_0) and C^N is the incidence matrix of (Q^N,M^N_0). So substituting in *eq.1* it yields that $C^D - B^T \varphi C = 0$ *(eq.2)*, such that $[1-B]^T[C^D,\varphi C]^T=0$ *(eq.3)*. So $[1-B]^T[M^D_k,\varphi M_k] = cte = 0$. As the model-diagnosis and model of the system are not synchronized, then the error e_k is calculated as: $e_k= M^D_k - B^T(\varphi M_k)$. When there is

not any fault, then $e_k = 0$, all columns of C^D are different between themselves and of the null vector, but when a fault occurs, $e_k \neq 0$, where the error is a column of C^D, is determined that a failure occurred. After the error is detected into the diagnoser it is necessary to know where it was the failure. In order to locate the failure it will be used the algorithm proposed in (Ruiz, 2007) that considers the risk transitions to determine if there exists a permanent failure or other kind of failures.

3.4 FAULT MODELS

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 and on the structure of the respective PN modules. Some faults that can be present into the system are control failure, permanent failure and intermittent failure. The first two kinds of failures are of interest for this work. A control failure represents the firing of a transition that exists in the system behavior, but should be avoided by the control system. A permanent failure occurs when a task stops its execution while other(s) task(s) can continue to run in the system. An intermittent failure is considered in the system when a task it runs out of its window of time set. These faults are modeled with the methodologies from (Ruiz, 2007) and are used to model the faults in the normal behavior of DES represented with IPN.

3.4.1 PERMANENT FAILURE MODELING

Given the model (Q, Mo) which describes the normal system, for each place p_i^N that represents an operation from which it can occur a failure, add an uncontrollable transition t_f, a place of failure p_j^N, and the arcs (p_i^N, t_f) and (t_f, p_j^N). The new place of failure p_i^F must be labeled with the same symbol p_i^N to represent that a failure cannot be detected from observing the system outputs (otherwise the detection would be trivial or immediate). T^{PF} is the permanent failures transitions set and P^{PF} is the permanent failures places set.

3.4.2 CONTROL FAILURE MODELING

Given the model (Q,Mo) which describes the normal system, for each place p_i^N that represents an operation from which a control failure f_c can occur, add a tamper-called transition t_{fc}, which must be connected to another place p_j^N (safe place that will be affected by the firing of the transition from control failure) and add the arcs (p_i^N, t_f) and (t_f, p_j^N). T^{CF} is the control failures transitions set and P^{CF} is the control failures places set.

4. CASE STUDY

Consider the following situation where there exists a water tank, in order to illustrate concepts of IPN and basic theory about the fault diagnosis problem. The system has a valve V and two sensors L and H to detect low level and high level respectively. Suppose that the level water is controlled by the valve: it can be open or closed. It is open when the water flow is no zero, in other case is closed. Also, the system has an output constant of water, and the output of each sensor is open if the water is in contact with them, in another case, is closed as it can see in the figure 3.

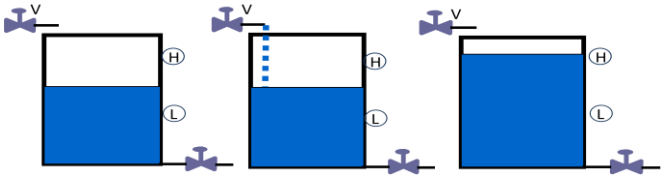

Figure 3: Water tank

4.1 SYTEM MODEL

The system is modeled using Petri net with Ramirez's methodology (Ramirez et al, 2007) that constructs a bounded and live system model; to use this methodology it is necessary to identify the components of the system, the variables range, and codifications of these variables in order to have a Petri net model for the valve and the

tank level. First, it is obtained the modules (Figure 4a) and after make compositions between them, it is obtained a final model of PN (Figure 4b).

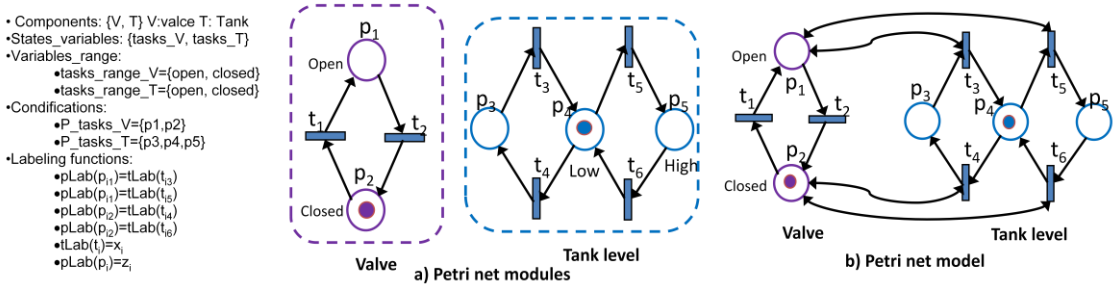

Figure 4: Model Petri nets

In this work is used the IPN so is necessary to establish labels in order to identify transitions and places that can be manipulated. For this case, it will be considerated $\lambda(t_1)$ =o, $\lambda(t_2)$=c, for others t, $\lambda(t_i)$= ε and open V event "o" and close V event "c". The places p_1, p_4, p_5 are measurable places. So the IPN model can be seen in the figure 5, with its output function and incidence matrix respectively.

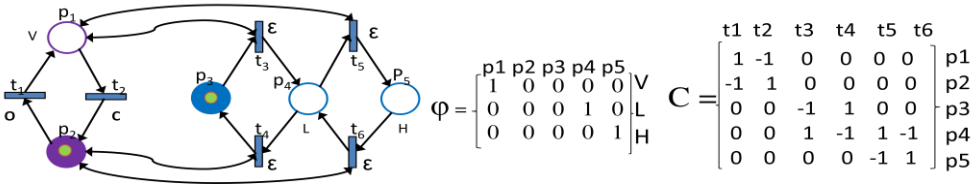

Figure 5: IPN model with its output function and incidence matrix

4.2 FAULT DIAGNOSIS

4.2.1 DIAGNOSABILITY TEST

The system needs to be reviewed, in order to know if the system pass the test of event-detectable, before being analyzed to see if it is diagnosable. It is considered the lemma mentioned above, in the diagnosability section. As it can see in the figure 6, the columns of the matrix φC are different between them and different from the null vector, so the system is event-detectable.

$$\varphi C = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

Figure 6: Matrix φC

Now, it can be modeled the faults in the system, as it can see in the figure 7. In this case, it is considered that permanent faults can occur when the valve is open or closed. The valve can stick. Also, it can exit a control fault when the water level changes from high level to low level. Always, it must be detected when the water level is a medium level.

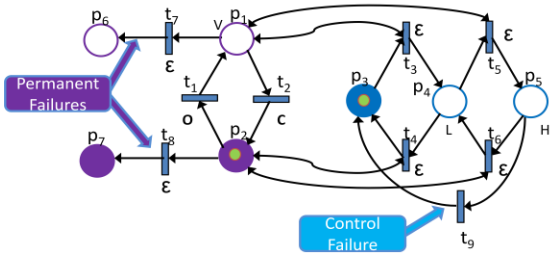

Figure 7: IPN modeling with normal and failure behaviors

After that, it is necessary to make the diagnosability testing, using the theorem from the section of diagnosability. As the $D_H(t_1,t_2)=1$ and both transitions just can have one mark into their input places, then the first step is passed. The normal behavior $(Q^N,M^N{}_0)$ of the fig above (6) is obtained using the Ramirez's methodology so the net is live and safe. Also, the permanent and the control faults are modeled with the Ruiz´s methodology and the lemma of the event-detectable is fulfill (the matrix of φC, only considers that the transitions of control faults are different), in the figure 8.

$$\varphi = \begin{array}{c} \begin{matrix} p1 & p2 & p3 & p4 & p5 & p6 & p7 \end{matrix} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array} \begin{matrix} V \\ L \\ H \end{matrix} \quad C = \begin{array}{c} \begin{matrix} t1 & t2 & t3 & t4 & t5 & t6 & t7 & t8 & t9 \end{matrix} \\ \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array} \begin{matrix} p1 \\ p2 \\ p3 \\ p4 \\ p5 \\ p6 \\ p7 \end{matrix} \quad \varphi C = \begin{array}{c} \begin{matrix} t1 & t2 & t3 & t4 & t5 & t6 & t7 & t8 & t9 \end{matrix} \\ \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & -1 \end{bmatrix} \end{array}$$

Figure 8: Matrix φC considering faults

4.2.2 THE DIAGNOSER MODEL CONSTRUCTION

The diagnoser model is constructed once that is known that the system is diagnosable. This is modeled with Petri nets too. In this case, it is constructed with one place considereng the Ruiz's methodoly. The diagnoser model $(Q^D, M^D{}_0)$ with $P^D=\{p_1,\ldots p_5\}$, and $T^D=\{t_1,\ldots t_6\}$) has a similar net than the normal behavior of the system, and has an incidence matriz C^D with a initial marking $M^D{}_o$.

Incidence matrix C^D is defined with the output function φ^N and the incidence matrix C^N of the normal behavior of the system, and a base vector B^T. So $C^D= B^T\varphi^N C^N$ is an incidence matrix of $(Q^D, M^D{}_0)$, C^N is the incidence matrix of $(Q^N,M^N{}_0)$ and $B^T = [b^0\, b^1 \ldots b^{q-1}]$, where $q= \#P_{measurable}$ and $b=2\max((abs(c_{ij}))+1$.

For this case: $b= 2\max(1)+1= 3$. The values of the C^N are 0,1 and -1, so $B^T = [b^0\, b^1\, b^2] = [1\ 3\ 9]$. The C^D and $M^D{}_o$ can be calculated. $C^D= [1\ 3\ 9]\ \varphi^N C^N = [1\ -1\ 3\ -3\ 6\ -6]$ and $M^D{}_o= [0]$. It can see in the next figure 9:

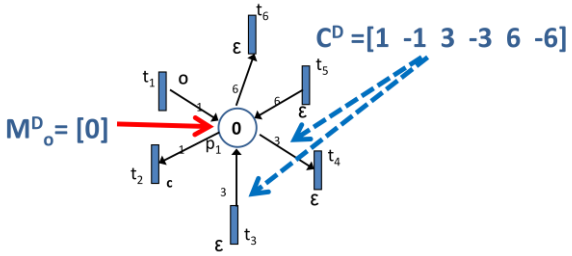

Figure 9: The diagnoser model

The values of the incidence matrix are the weights of the arcs, positives values of the incidence matrix are arrows entering and the others are the arrows coming out of the initial marking. This diagnoser model is used into the diagnosis scheme in order to detect and locate faults.

4.2.3 DETECTION OF FAULTS

The diagnosis scheme used the diagnoser model, that contains the normal behaviour of the system, to compare it with the anormal behaviour of the system and calculates an error. The error helps to identify if there exist a fault. If the error is equal zero means that there no exist a fault but if the error is different from zero, means that there exist a fault. The error is defined by $e_k=M^D{}_k-B^T\varphi(M_k)$, as it was defined above in the diagnosis scheme section. $M^D{}_k$ is the marking of the diagnoser model and $\varphi(M_k)$ is the observation vector of the system model.

If it is considered a transition sequence $\sigma= t_1t_3t_5$, it can see that the ouput of the system model has the observation vector $(\varphi(M_k))^T= [1\ 0\ 1]$ (the marks that exist into the places that were activated with these transitions) and the marking of the diagnoser model is $M^D{}_k = 1 + 3 + 6 + 0 = 10$ (the wieght of the arcs of the transitions selected with

M^D_o). The error is e_k=10 - [1 3 9]^T [1 0 1] = 10 - 10. Now, if after the transition sequence σ= $t_1t_3t_5$, is activated a permanent fault t_7 the place p_6 is marked but this is no detected because there no exist into the system model, the observation vector $(\varphi(M_k))^T$= [1 0 1] and the diagnoser model does not change its marking (M^D_k = 1 + 3 + 6 + 0 = 10). If after that, the transition t_2 is fired, then the diagnoser model changes its marking M^D_{k+1} = 1 + 3 + 6 + 0 - 1= 9 but the observation vector continues being the same $(\varphi(M_k))^T$= [1 0 1] so the error is different from zero e_k= [9]-[10]. Here is detected a fault, the permanent fault. The output system is the same but the output of the diagnoser model has a new value, the weight of arc of the transition t_2. So, the failure is detected.

5. CONCLUSIONS

Here was presented a case study to understand the process of the fault diagnosis for DES, from (Ruiz et al, 2007). Here is considered how to construct the system model with IPN such as the diagnoser model with PN. It is necessary to have a diagnosis scheme, that uses the normal behavior model and its modeled of failures. Then, the system must pass the analysis of the diagnosability property. After that, it is made the design of diagnosis that considers two important steps: 1) a diagnoser model and 2) an error calculation. After analyze that DES complies with the diagnosability property (using the Ramírez´s methodoly and using definition of event-detectablility), it can be constructed the diagnoser model using the Ruiz's methodology. As future work it is considered to modify this diagnoser model to include more kinds of DES (topologies with different of the net of free election), and redefine the diagnosability property.

REFERENCES

Arámburo J. (2009). "Reliable Distributed Diagnosis in Discrete Event Systems" Ph.D. thesis, CINVESTAV-unidad Guadalajara, México. pp. 1- 68.

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

Chung L. C. (2005). "Diagnosis PN-based models with partial observable transitions". *International Journal Computer Integrated Manufacturing*, vol. 18, pp. 158-169.

Debuck R. (2003). "Diagnosis of Discrete Event System. A modular Approach". *Proceedings of the IEEE Conference on System, Man and Cybernetics*, pp. 306-3011.

Genc S. and Lafortine S. (2007). "Distributed Diagnosis of Place-Bordered Petri Net". *IEEE Transaction on Automation Science and Engineering*. Vol. 4, No. 2 pp. 206-219.

Hadjicostis C. N. (2002). "Probabilistic Fault Detection in Finite-State Machines Based on State Occupany Measurements". *Proceedings of the 41st IEEE Conference on Decision and Control*, vol.4, pp. 3994-3999.

Lafortune S., Teneketzis D. and Sampath (2001). "Failure Diagnosis of Dynamic Systems: An approach based on Discrete Event Systems". *Proceedings of the American Control Conferen*, pp. 2058-2068.

Murata T. (1989). "Petri nets: properties, analysis and applications". *Proceedings of IEEE*, vol. 77 no. 4, pp. 541-580.

Peterson J. L. (1981). "Petri Net Theory amd Modeling of Systes". Prentice-Hall.

Rivera-Rangel I., Ramírez-Treviño A., Aguirre-Salas L.I. and Ruiz-León J. (2005). "Geometrical characterization of Observability in Interpreted Petri Nets". Kybernetika, vo.41, 553-574.

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

Ramírez-Treviño A., Ruiz-Beltrán E., Rivera-Rangel I. and López-Mellado E. (2004). "Diagnosability of Discrete Event Systems. A Petri Net Based Approach". *Proceedings of the IEEE International Conference on Robotic and Automation*, pp. 541-546, 2004.

Ruiz B. E. (2007). " Diagnostic Diagrams Discrete Event Systems". Ph.D. thesis, CINVESTAV-unidad Guadalajara, México. pp. 1- 150.

Ruiz B. E., Jiménez O. I. Ramírez T. A. López Mellado E. and Meda C. M. (2005). "Fault detection and location in DES modeled using Petri Nets". *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, pp. 1645-1650.

Ruiz B. E., Ramírez T. A. and López Mellado E. (2004). "Building Diagnosable Petri Net Models for Distribuited Fault Location of DES". *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, pp. 4929-4933.

Sampath Meera, Sengupta Raja, Lafortune Stephane, Sinnamohideen Kasim, and Teneketzis Demosthenis C. (1996). Failure Diagnosis Using Discrete-Event Models. *IEEE Transactions on Control Systems Technology* Vol. 4, No. 2, pp.105-124.

Sampath M., Lafortune S., Sinnamohideen K. and Teneketzis Demosthenis C. (1998). "Diagnosis of Discrete-Event Systems", *IEEE Transactions on Automatic Control*, Vol. 43, no. 7, pp. 908-929.

Silva M. (1985). Las redes de Petri: en la automática y la informática. Editorial AC, Madrid, España.

Ushio T., Onishi I. and Okuda K. (1998). "Fault Detection Based on Petri Net Models with Faulty Behaviors". *IEEE Transactions on Automatic Control*. Vol. 50, no. 4, pp. 476-492.

Xiaoli1 Wang, Chen Guangju Xie yue Guo Zhaoxin (2009). "Fault Detection and Diagnosis Based on Time Petri Net", *The Eighth International Conference on Electronic Measurement and Instruments*, pp.3-259 - 3-26.

Authorization and Disclaimer

Authors authorize LACCEI to publish the paper in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.