DIAGNOSIS AND TREATMENT OF FAULTS IN MANUFACTURING SYSTEMS BASED ON BAYESIAN NETWORKS AND PETRI NETS

Roy Andres Gomez Morales, roy.morales@poli.usp.br
Paulo Eigi Miyagi, pemiyagi@usp.br
Escola Politecnica – University of São Paulo

Luis Alberto Martinez Riascos, luis.riascos@ufabc.br
Federal University of ABC

Abstract. *Concerning with the new paradigms referred to manufacturing systems, new challenges for the control and monitoring processes include the early detection, diagnosis and treatment of incipient faults. Such systems have been beneficed by the last decade computational revolution; however, this also has brought the necessity of consider problems of uncertainty in the information sharing, the distributed control, coordination and cooperation of autonomous entities, remote operation and so on. For those reasons, all these tasks have to be performed concerning not only the normal behavior of the system, but also the behavior in abnormal (faulty) conditions. In this context, diagnostic processes obtain the causes of failures given a set of observations and treatment processes include the decision of the best procedure for the total or partial system recovery. Petri net has been proved suitable for the modeling of control systems, and Bayesian networks uses probabilistic relationships among its variables making them suitable for inference purposes. Therefore, this works aims for the integration between Bayesian networks for the diagnostic and decision making processes and Petri nets for the control process. A hierarchical approach is developed for the modeling stage. A case of study which consists of a single line manufacturing system has been considered in order to evaluate the procedure. Special emphasis is focused on methodological issues and industrial systems.*

***Keywords**: Manufacturing systems, Discrete Event Systems, Petri Nets, Bayesian Networks, Decision Theory.*

1. INTRODUCTION

In the past few years, manufacturing systems have experimented many changes in terms of complexity based on the customers' constantly changing demand and volatility. New challenges, especially for emergent countries since competitiveness demand products with more quality at lower costs, have concerned many researches (Beach et al., 2000), (Abell, 1979), (Spina et al., 1996) and the key points are the improvement of effectiveness of strategies in planning, scheduling, coordination, and control of such systems (Santos Filho, 2000). Besides, we detach a critical aspect in manufacturing process that is: faulty behavior of components, machine breakdowns and device failures. Concerning with such aspects can make the difference between a successful system and an ailing one. For that reason, the control and monitoring processes in a production system must include the early detection, diagnosis and treatment of incipient faults.

In this context, Petri net has been proved useful in the modeling, analysis and implementation of manufacturing control systems (Crockett, et al., 1987), (Desrochers and Al-Jaar, 1995), (Zhou and Venkatesh, 1999). In detection techniques, Riascos et al. (2004) propose the use of such net for the detection and treatment of faults in manufacturing systems, and Lira (2005) introduces the detection of faults in assembly systems by the use of Petri net.

On the other hand, Bayesian networks use probabilistic relationships among their variables to represent influences between each other (Pearl, 1988), (Neapolitan, 2003). From such relations, the model permits making inferences, which means that from a set of observed variables, the most probably explanation for such evidence can be obtained. Przytula and Thompson (2000) introduce the construction of Bayesian networks for diagnostic processes and Riascos et al. (2003) presents the detection and treatment of faults based on Petri nets and Bayesian networks.

The present work introduce the use of Petri nets in the modeling stage of the control system and the use of Bayesian networks for the diagnostic and decision making processes. Since the use of Petri net for modeling of control systems usually requires a large amount of nodes, a hierarchical approach is proposed in order to avoid complexity in the representation.

This paper is organized as follows. Section 2 introduces the general concepts of Petri net and Bayesian networks. Section 3 presents the proposed methodology. In section 4 a single line manufacturing system is considered as a case of study in order to validate the proposal. Finally in section 5 the conclusions of the work and the open topics for research are presented.

2. BACKGROUND ON PETRI NETS AND BAYESIAN NETWORKS

This section introduces the general concepts of Petri net and Bayesian networks, further readings include (Murata, 1989), (Peterson, 1981), and (Reisig, 1985) for Petri net theory, and (Pearl, 1988), (Pearl, 2000), (Neapolitan, 2003) and

(Jensen, 2001) for Bayesian belief networks theory. A text book in applications of Petri net in hierarchical modeling and control of discrete-event systems can be found in (Miyagi, 1996).

2.1. Petri net

Petri net is a technique for the synthesis, modeling, analysis and implementation of systems. It works with process modeling, which is a representation of what we believe are the most important features of it. Such a model consists in a graphical structure and a mathematical support that permits the analysis of the system in order to obtain information relevant for the correct behavior of the components.

Formally, a Petri net is a bipartite directed graph which consists in a set of places $P = \{p_1, \ldots, p_m\}$ represented by circles; a set of transitions $T = \{t_1, \ldots, t_n\}$ represented by rectangles; a function of income arcs $I = P \times T \rightarrow \mathbb{N}$ represented by arcs from places to transitions; a function of outcome arcs $O = T \times P \rightarrow \mathbb{N}$ represented by arcs from transitions to places; and an initial marking $M_0 : P \rightarrow \mathbb{N}$ that assigns to each place a number of tokens, and that represents the state of each place at any given time.

Places in manufacturing environment, can represent operations or processes, availability of resources, or availability of raw materials, fixtures, pallets, etc. Transitions represent activities that imply a change of state such as starting/stopping of an activity and that distribute the resources to the given processes.

The dynamic behavior of a system is modeled in Petri net by the enabling and firing of transitions. The transition firing rules are:

i) a transition t is said to be enabled *iff* $m(p) \geq I(p,t)$, where: $m(p)$ is number of tokens in each input place of t and $I(p,t)$ is the number of input arcs from p to t (also called weight of the arc);

ii) an enabled transition may or may not fire, depending on the evolution of the system; and

iii) when a transition fires, a new marking is defined as: $M_i(p) = M_{i-1}(p) - I(p,t) + O(t,p)$, where $M_i(p)$ is new marking of p, $M_{i-1}(p)$ is the marking before t was fired, and $O(t,p)$ is the number of arcs to p.

In control of manufacturing systems based on Petri net, three important properties are required to be maintained (Zhou et al., 1992): boundedness, liveness and reversibility.

Boundedness is the property that guarantees that any given place will no have more than a k number of tokens, i.e., $m(p) \leq k$, $\forall p \in P$, where k is an integer such that $k \geq 1$; liveness is the property that guarantees that for any given transition t, there is a sequence of firings that will enable it; finally, reversibility is the property that guarantees that for any marking $M_i(P)$, there is a sequence of firings such that it will lead the system to the initial state $M_0(P)$. A special case of boundedness is safeness where $k = 1$.

Since the modeling of manufacturing systems generally results in a large amount of nodes, comprising the representational power of Petri net, the modeling of the control system is divided in two levels: a coordination level and a plant control level. The coordination level is a safe, live and reversible Petri net that represents the evolution of the process under study. The control level contains the operations, availability, and behavior of the components of the process. Therefore, a top-down approach can be implemented starting with the construction of the coordination level, and making a refinement in order to obtain the plant control level.

Murata (1989) offers various analysis techniques for the verification of the properties mentioned above, that include the reachability graph, incidence matrix, siphons and traps, etc. Also, simulation can be used in order to verify the properties of the model.

2.2 Bayesian Networks

Bayesian belief networks (Pearl, 1988) are structures that represent probabilistic influences among their variables. Informally, the construction of a Bayesian network can be performed by taking one node and linking to it all other variables that have an influence on it or that it has influence to.

Bayesian methods provide a formalism for reasoning about partial beliefs under conditions of uncertainty. In this formalism, propositions are given as numerical parameters signifying the degree of belief according to some evidence or knowledge. So, formally, Bayesian networks $(G, Prob)$ are constituted by a topological structure G and a set of parameters $Prob$ which represents the probabilistic relations among their variables.

The structure of Bayesian networks are represented by directed acyclic graphs (*DAG*) where each node X_i has a set of parents $pa(X_i)$, a set of children $ch(X_i)$, and a set of children's parents (also called spouses) $spo(X_i)$. The parameters of the nodes are given by the conditional probability of each variable assume a value x_i, given the observation of the set of its parents, i.e.,

$$Prob(X_i \mid pa(X_i)) \tag{1}$$

Normally, such parameters are given in a table format for which they are also called conditional probabilistic tables (CPT) (Murhpy, 1998).

At this point it is necessary to define the concept of probabilistic models. A probabilistic model is an encoding of probabilistic information used to compute the probability of every well-formed sentence *S* (a set of variables) in accordance with the basic principles of probability (Pearl, 1988). So, any joint probability function represents a complete probabilistic model. In this sense, in Bayesian networks the computing of joint probability distributions can be suitable obtained by the calculation of:

$$Prob(x_1, x_2, ..., x_n) = \prod_{i=1}^{n} P(x_i \mid pa(x_i)) \tag{2}$$

One advantage of using Bayesian networks is that they permit in a suitable way, represent human knowledge explicitly, directly and qualitatively in a graph. However, in modeling conceptual relations, such as causation, association, and relevance, it is often hard to distinguish direct neighbors from indirect neighbors; then, constructing a graph for these relations becomes a delicate task.

The construction of a Bayesian network can be performed in the following ways

- based on knowledge elicitation about the system, where relationships among the variables are established;
- based on system manuals, handbooks and human knowledge about the process; or
- based on learning methods using databases of records of past operations.

Since the construction of Bayesian networks based on learning methods requires a large amount of records of past system failures, in this paper a combination of the two former methods is implemented.

One motivation for the use of Bayesian networks in this work is that they can be useful for inference purposes. In general, probabilistic inference on a belief Bayesian network is the process of computing $Prob(X = x \mid E = e)$, which obtains the probability that a variable *X* will assume a value of *x* given that a set of observations *E* have assumed a set of values *e*.

There are many different algorithms for calculating the inference in Bayesian networks, which apply different tradeoffs among speed, complexity, generality, and accuracy. In this work it is implemented the probability propagation in trees of clusters (PPTC) algorithm, developed by Lauritzen and Spiegehalter (1988), which is an exact method of inference and generally, can be applied to any type of Bayesian network structure (Cozman, 2001).

3. PROCEDURE FOR DIAGNOSIS AND TREATMENT PROCESS

The general procedure for the diagnosis and treatment of faults in manufacturing systems proposed here, has the following steps:

1. Construction of the process model based on Petri net.
2. Classification of places in the Petri net model based on the definition mentioned in section 2.1.
3. Construction of the Bayesian network(s).
4. Definition of the best treatment process considering each type of fault.

3.1. Construction of the process model based on Petri nets

As mentioned before, the construction of the Petri net starts with the coordination level, which is a safe, live and reversible Petri net. In this context, each place may represent a certain state of the system, a process or a condition; transitions may represent the beginning or the end of the initial operations of each process, or simply a change on the state of the process. In this model, the properties of liveness, safeness and reversibility must be verified using one of the techniques mentioned in previous section. Also, this net is responsible for the resolution of conflicts and contacts in the plant control net (Murata, 1989) (Combacau and Courvoisier, 1990).

Based on this model, a refinement can be made in order to obtain a more detailed net that represents the correct functioning of the system. This net represents the plant control level and, depending on the system, can contain contacts and conflicts. This two system’s characteristics must be solved in the coordination level, so a refinement in the original net is also required. The plant control net that represents the control of a manufacturing system generally will tend to present mutual exclusion operations, so shared resources can result in a state of deadlock. A good reference to avoid deadlock states can be found in Nakamoto (2003). Basically, an adequate administration of the shared resources will avoid a deadlock possibility.

Note that knowledge of the dynamics of the system is required in order to construct a reliable model. Thus, a model validation can be based on scenarios simulation observing if the net evolution represents the actual behavior of the system. This is done also in order to analyze and improve the model reliability. Software packages like HPSim (2001) and Visual Object Net ++ (VON) (2003) are examples of tools for such simulation purposes.

3.2. Classification of Places

In order to get the set of variables that will feed the Bayesian network as evidence, a classification of places on the Petri net have to be made based on the characteristics of each place. For this task, places in the plant control net are classified in places of type *A* and places of type *B*. Places of type *A* represent operations, process or resources (e.g., starting/ending of a machine process, starting/ending of a robot operation, state of buffers, etc.) and because of its nature, they normally have a null initial marking, i.e., $M_0(P_A) = 0$, $\forall P_A \in P$. On the opposite, places of type *B* represent states of the sensors and actuators that are implemented in the real system. This type of places have the characteristic that are safe due to its digital functioning (i.e., "on" or "off" states). Note that, on the opposite of places of type *A*, some *B* places can actually have one initial token, i.e., $\exists p \in P_A$ such that $m_0(p) = 1$.

3.3. Construction of the Bayesian network

Bayesian networks constitute a very attractive tool for diagnostic purposes since they can represent the diagnostic domain in conjunction with system's components and results of diagnostic tests. Nevertheless, the construction of Bayesian networks is a complex task, involving the participation of knowledge engineers and domain experts, with additional knowledge coming from such sources as technical manuals, test procedures, and repair data bases (Przytula and Thompson, 2000). Two steps involve the construction of Bayesian networks: the creation of the structure and the elicitation of the conditional probability parameters over the variables.

The first task for the creation of the Bayesian network structure is the definition of the number of faults that the system will be designed for. Based on this number, a classification in large and simple systems can be done. Each of these faults are represented by a variable with two possible states, i.e., the occurrence or not occurrence of the fault. Once defined the set of faults, is necessary to obtain the possible influences that each fault has into some components of the system. Thus, a set of variables representing system's components will be obtained based on the observation of such influences.

Normally, the complexity of this task depends on the size of the system. For large systems, decomposition on simple subsystems will facilitate the structure formulation improving the reliability of the model. Decomposition in simpler structures can be done based on divisions of the process or based on the way which the diagnosis is actually executed in practice. Normally, is used a combination of both of these approaches depending on each subsystem complexity.

Once a subsystem is defined, it is necessary to obtain the information about the subsystem functions and how this subsystem fails. In general, such information is available from different sources like manuals, testing procedures, and most important expert's knowledge. Przytula and Thompson (2000) proposes a ranking of the faults defined for each subsystem based on the frequency of occurrence, in order to group the faults that happen occasionally in only one variable such as "other faults".

Once the network topology is obtained, it is necessary to obtain the parameters of each node. The parameters can be assessed in the causal direction, i.e., the probability that a given evidence will be observed once is known that a given component is in a faulty condition. Prior probabilities of roof nodes, i.e., nodes without parents, can be assessed based in statistical information of the components they represent.

Sometimes, causal elicitation of probabilities is a difficult task, and rather is possible to obtain the parameters of the nodes in the diagnostic way, i.e., the probability that a given component is in a faulty state, given that a set of evidence has been observed (i.e., $Prob\left(C = fail \middle| E = observed\right)$). Maintenance engineers normally found this way more feasible. It has been proven that, the joint probability distribution of any model with diagnostic parameters is completely defined if the information of the probability of a fault occurrence given that a set of evidence is inactive, is also available (i.e., $Prob\left(C = fail \middle| E = inactive\right)$) (Przytula and Thompson, 2000).

Figure 1 presents an example of a simple Bayesian network representing two types of faults (*F1*and *F2*) and a set of four observations (*E1, E2, E3, E4*). The figure also includes an auxiliary node (*Aux*) that represents some variable that cannot be observed directly from the system, but has a causal influence on an evidence or a symptom.

It is important to note that the construction of Bayesian networks is an iterative task, refining the model in each interaction based on new evidence observed.

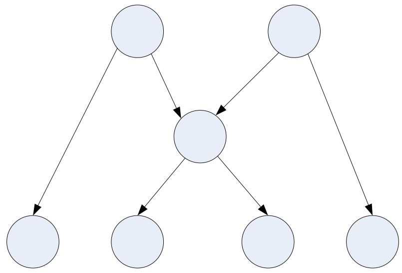

Figure 1. A simple Bayesian network representing two faults, a set of observations and an auxiliary node.

3.4. Definition of Treatment process

Zhou and DiCesare (1993) present a classification and modeling of treatment of faults processes based on Petri nets. In this context, four types of treatments are defined: Conditional entrance method, alternative route method, inverse recovery and direct recovery methods.

In the conditional entrance method, a corrective function is triggered once a fault is detected. Figure 2(a) represent the model of this method where **Z** is the original net and **S** is a subnet representing the fault recovery procedure. In the alternative route method the fault operation **Q** will be avoid deviating the sequence of the process by a sub-process **S**, represented in fig. 2(b). In the inverse recovery method, after a faulty execution of a **Q** process is observed, a sequence **S** will lead the system to a re-execution of **Q** (Fig. 2(c)). Finally, in the direct recovery method, detecting a fault in the process **Q** will lead the system to execute a sequence **S**. This method is represented in Fig. 2(d).

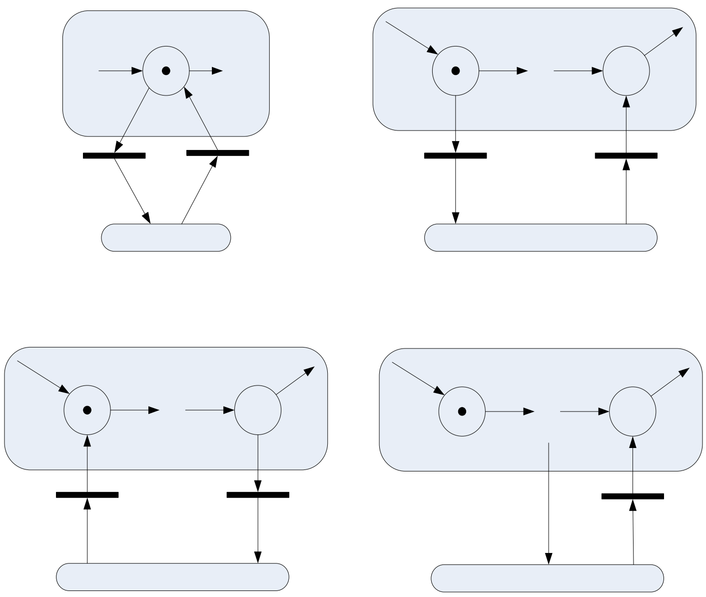

Figure 2. Representation of treatment processes: *(a)* conditional entrance method, *(b)* alternative route method, *(c)* inverse recovery and *(d)* direct recovery method.

4. CASE OF STUDY

This chapter introduces a case of study used to illustrate the application of the proposed methodology. It consists in a manufacturing station constituted by a flexible assembly system commercialized by FESTO. This system is illustrated in Fig. 3. The goal of the process is to produce an assembled component that consists in a base, a pine, a spring and a cover as presented in Fig. 4. Bases can be of three colors: black, pink or metal color and, depending on such color, pines assembled on it can be black or metal. If the base is color pink or metallic, then the pine must be black color; on the opposite, if the base is black, the pine must be metallic.

The system consists in five stations: a material distribution station, a testing station, an intelligent transportation system, an assembly station and a control system. The material distribution station contains the bases stored in random colors, a cylinder that accommodates the base for the transportation and a 180° arm that deposits the base at the testing station. The testing station consists of three sensors: an inductive, a capacitive and a photoelectric, in order to detect the color of the base. Also it consists of a vertical transportation system, a measurement sensor in order to verify the size of the base and a cylinder that pushes the base to the transportation system. The transportation system consists in moving pallets with four stops, one in the beginning of the circuit, another in the output of the testing station; the third in the assembly station and the fourth at the output of the circuit. Each of these stages are monitored by presence sensors that detect the presence of pallets, responsible for the housing of the bases and the finished products. Finally, the assembly system is constituted by a pine storage, spring storage, a cover storage and a robot responsible for the assembly functions.

All these systems have their own set of sensors and actuators which description, for reasons of space, is omitted in this text.

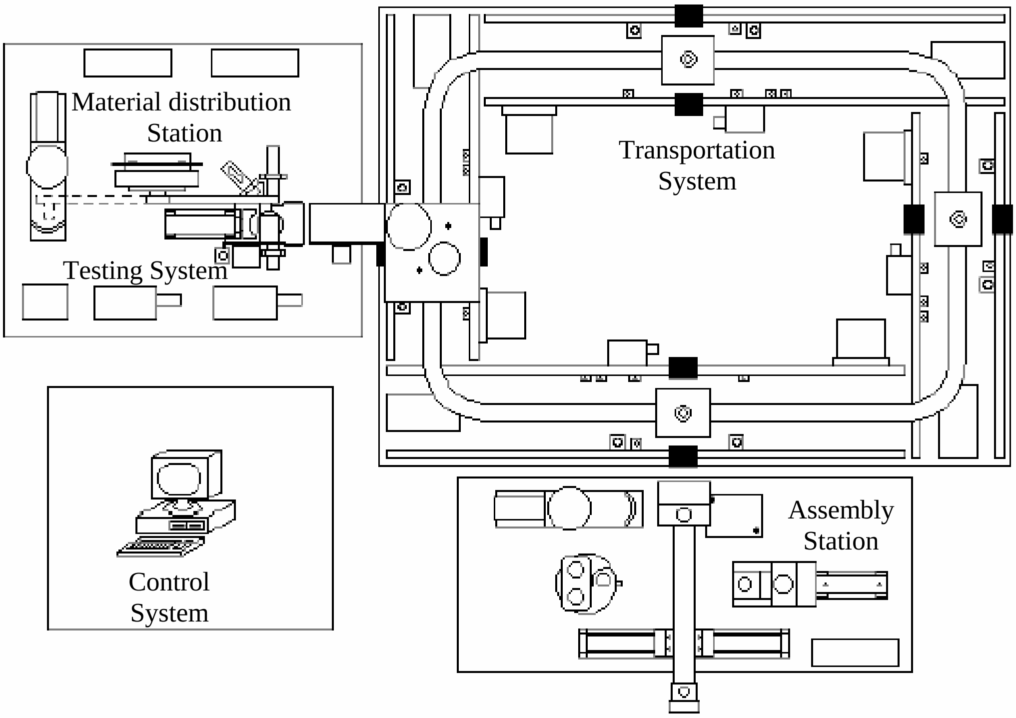

Figure 3. Scheme of the manufacturing system station.

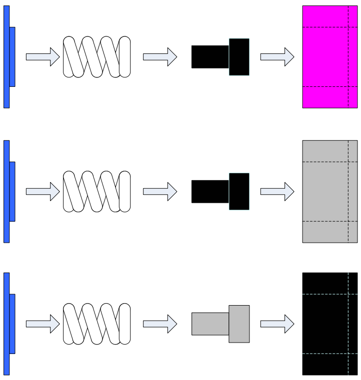

Figure 4. Sequence of the assembly for the final product of the assembly station.

4.1. Petri net representation of the system

The coordination level Petri net is presented in Fig. 5. The resulting net, in this particular case, is an acyclic choice Petri net or AC net (Murata, 1989). Using the siphons and traps technique, it can be proved that this net is life, safe and reversible. Note that the mark of the place "distribution" depends on the user's requirement of a production, so corresponded transitions are guaranteed with liveness.

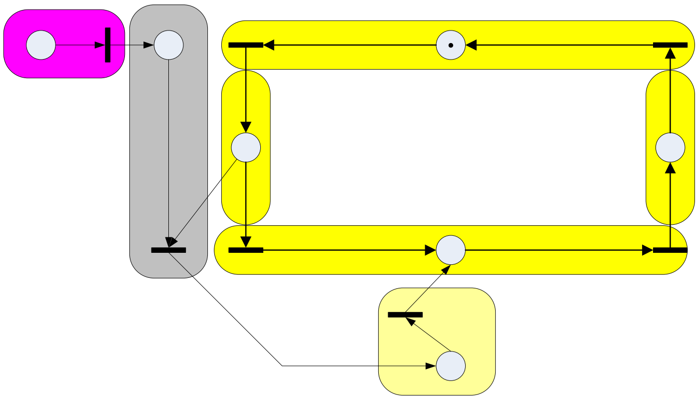

Figure 5. Coordination level Petri net.

Based in this net, a refinement is developed to detail the plant control level. The resulting net describes in a more detailed manner the functionality of the system, showing the behavior of components in each correspond station. It also represents the sensors and actuators that operate in each station, showing their functional behavior based on the evolution of the process (Fig. 6).

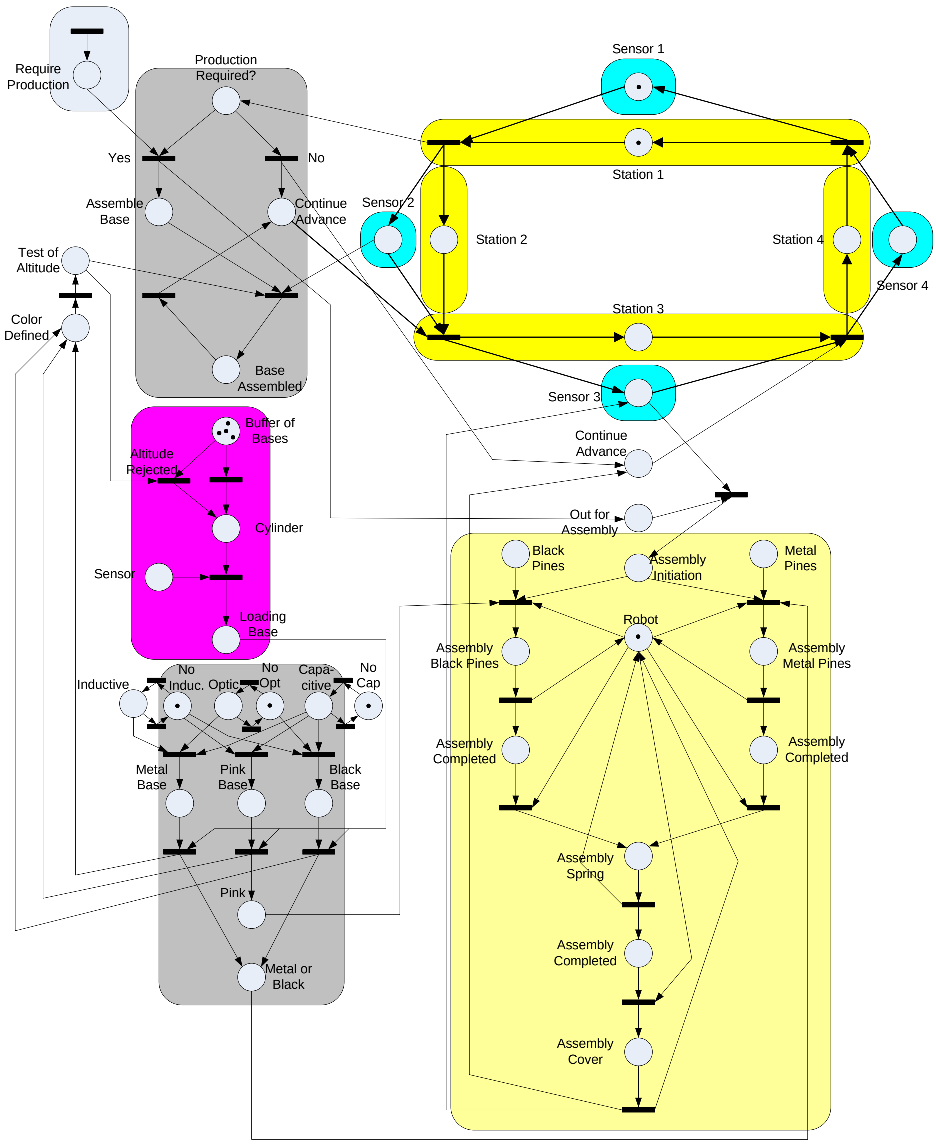

Figure 6. Plant control level Petri net.

4.2. Classification of places

As mentioned in section 3.2, a classification of places based on the functionality of the nodes is then conducted. Figure 6 shows places of type *A* and places of type *B*, other places represents availability of raw material. Table 1, shows the list of the places of type *B*.

Table 1. Places of type *B*.

NAME	FUNCTION
Require Production	User sends a message that requires an assembly
Sensor	Detects the presence of a base
Sensors 1…4	Detect the presence of a pallet in the station 1…4
Cylinder	Pushes a base for start the loading
Test Altitude	Measures the altitude of the base
Loading Base	Place the base in the test station
Inductive	Detects if the base is metallic
Optic	Detects if the base is not black
Capacitive	Detects if the base is not metallic

4.3. Construction of the Bayesian Network

The construction of the Bayesian network it derived from an exhaustive observation and analysis of the system's behavior, based on the functioning of the components. Some fails have been forced in order to observe their effects on the evolution of the sequence of the process. Such forced faults consist in disconnecting some sensors, or deactivating some actuators. The resulting list of faults is showed in Tab. 2.

Table 2. Faults considered for the construction of the Bayesian network.

FAULT PATTERN	SYMPTOM
Fault in the conveyor system	The pallet stays in one station, causing The overall stop of the process
Fault in the distribution system	A base is not loading, causing The overall stop of the process
Fault in the color recognition	A color is wrongly recognized, causing wrong assembly or overall stop
Fault in the altitude test	Too much bases are rejected, or no detection is made at all

Based on these faults and the set of observations, the Bayesian network obtained is illustrated in Fig. 7.

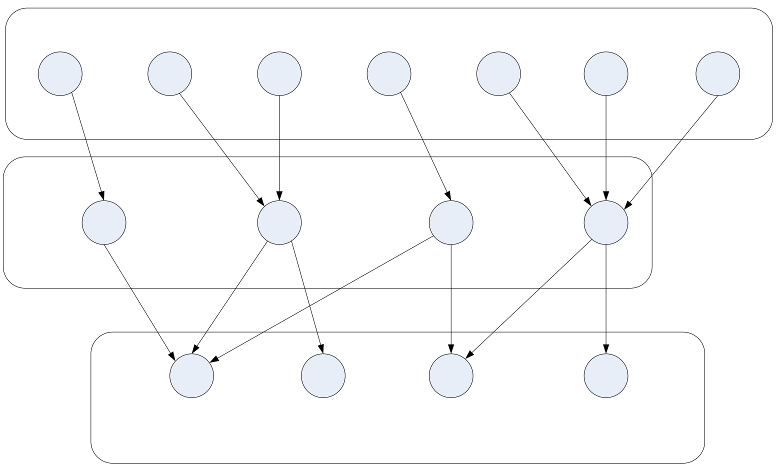

Figure 7. Bayesian network for the diagnostic of faults.

4.4. Definition of the treatment processes

In order to illustrate a treatment process, an alternative route method is implemented in the size test system. Once we know that a fault has occurred with the size sensor, the sequence of the system can be modified avoiding the activation of such component. This is done in order to avoid interruption of the process, until maintenance is implemented in the device. In this context, Fig. 8 introduces the alternative route for such types of faults. **S** represents the alternative route that avoids the faulty component.

Unfortunately, not all the components of the system under study permit the implementation of an automatic treatment and recovery of the process under faulty conditions. However, an inverse recovery method can be implemented, in manual way, reestablishing the components under faults and making a corrective maintenance of the components listed as faultily.

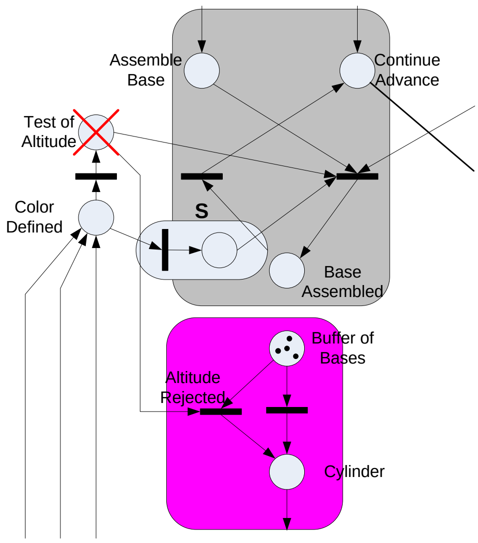

Figure 8. Alternative route for altitude test faults

5. CONCLUSIONS

The present work introduces a methodology for the diagnosis and treatment of faults in manufacturing systems based on Petri net and Bayesian networks. The Petri net representation begins with the coordination level, representing in each node, macro-activities or operations. The basic idea is that this net only represents the correct sequence of system operation. Also, conflicts and contacts must be solved in the net based on specifications of the system. From the coordination level, a refinement is developed to detail the plant control. This net usually contains a large amount of nodes, so a classification of places is made in order to obtain the set of evidence for which the Bayesian network will interpret observations.

The Bayesian network is obtained, first constructing the structure of the net and second, obtaining the probabilistic parameters in a causal way or in a diagnostic way of each node. Inference based on this model has shown that the diagnostic is faithful with reality, pointing out always the component responsible for the faulty behavior of the system.

Treatment process includes conditional entrance method, alternative route method, inverse recovery and direct recovery methods. Depending on the type of fault one of these methods can be implemented in order to execute automatic (total or partial) recovery of the system. However, some cases will enforce the system to run a corrective maintenance procedure.

Open research issues include the integration of the diagnostic and treatment process with higher levels of manufacturing control, i.e., scheduling and planning level. This integration is fundamental to improve the flexibility and agibility of planning and scheduling under faults that disturbs the correct functioning of the process.

6. ACKNOWLEDGEMENTS

This work was supported in part by the Brazilian governmental agencies CNPq, CAPES and FAPESP.

7. REFERENCES

Abell, D.F., 1979, “Defining the bussiness”, Prentice-Hall, New York, NY.

Beach, R., Muhlemann, A.P., Price D.H.R., Paterson, A., and Sharp, J.A., 2000, “Manufacturing operations and strategic flexibility: survey and cases”, International journal of operations & production management, vol. 20, No. 1, pp. 7-30.
Combacau, M., and Courvoisier, M., 1990, “A hierarchical and modular structure for FMS control and monitoring” Proceedings in AI, Simulation, and Planning in High Autonomy Systems, pp. 80-88.
Cozman, F.G., 2001, <http://www-2.cs.cmu.edu/~javabayes/ >
Crockett, D., Desrochers, A., DiCesare, F., Ward, T., 1987, “Implementation of a Petri net controller for a machining workstation”, Proceeding in Robotics and Automation, Vollume 4, pp. 1861-1867.
Desrochers, A.A. and Al-Jaar, R.Y., 1995, “Applications of Petri nets in manufacturing systems”, Modeling, Control and Performance Analysis, IEEE Press, pp. 348.
HPSIM, Version 1.1, 2001. Universitaet Bern: Copyright (c), Henryk Anschuetz, <http://www.winpesim.de/petrinet/e/hpsim_e.htm>.
Jensen, F., 2001, “Bayesian Networks and Decision Graphs”, Springer.
Lauritzen, S.L., and Spiegelhalter, D.J., 1988, “Local computations with probabilities on graphical structures and their application to expert systems”, J. Roy. Stat. Soc. B, 50, 157-224.
Lira, D., 2005a, “Detecção de falhas em sistemas flexiveis de montagem através de rede de Petri”, Dissertação de Mestrado, Escola Politécnica da Universidade de São Paulo, São Paulo, SP.
Lira, D., Junqueira, F and Miyagi, P.E., 2005b, “Process Modeling and Fault Detection of Flexible Assembly Systems using Petri net”, 18th Intern. Congr. of Mechanical Engineering, COBEM 2005. ABCM, Ouro Preto, Brazil.
Miyagi, P.E., 1996, “Controle Programável – fundamentos do controle de sistemas a eventos discretos”, São Paulo, Editora Edgard Blucher.
Murata, T., 1989, “Petri Nets: Properties, Analysis and Applications”, Proceeding of the IEE, Vol. 77, No. 4, pp. 318-328.
Murphy, K., 1998, “A Brief Introduction to Graphical Models and Bayesian Networks”, 22 February 2007, < http://www.cs.ubc.ca/~murphyk/Bayes/bayes.html>
Nakamoto, F.Y., 2002, “Sistematização do Projeto do Controle de Sistemas Produtivos”, Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do titulo de Mestre em Engenharia.
Nakamoto, F.Y., Maruyama, N., Miyagi, P.E., Santos Filho, D.J., 2001, “Geração automática de regras de controle para alocação de recursos em sistemas produtivos com processos concorrentes”, 16th Intern. Congr. of Mechanical Engineering, COBEM 2001. ABCM, Ubêrlandia, Brazil
Neapolitan, R., 2003, “Learning Bayesian Networks”, London: Pearson Printice Hall.
Pearl, J., 1988, “Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference”, Morgan Kaufmann Publishers, San Mateo, California.
Pearl, J., 2000, “Causality: Models, Reasoning, and Inference”, Cambridge University Press.
Peterson, J.L., 1981, “Petri Net Theory and The Modeling of Systems”, Englewood Cliffs, Prentice-Hall.
Przytula, K.W., Thompson, D., 2000, “Construction of Bayesian networks for diagnostics”, Aerospace Conference Proceedings, Vol. 5, pp. 193-299.
Reisig, W., 1985, “Petri Nets an Introduction”, New York, Spring Verlag.
Riascos, L.A.M., Cozman, F.G., Miyagi, P.E., 2003, “Detection and Treatment of faults in automated machines based on Petri nets and Bayesian networks”, In. International symposium on industrial electronics (ISIE), Rio de Janeiro.
Riascos, L.A.M., Moscato, L.A., Miyagi, P.E., 2004, “Detection and treatment of faults in manufacturing systems based on Petri Nets”, Journal of The Brazilian Society of Mechanical Sciences and Engineering, ABCM, Rio de Janeiro, vol. 26, no. 23, pp. 280-289.
Santos Filho, D.J., 2000, “Aspectos do Projeto de Sistemas Produtivos”, tese de livre docência, Escola Politécnica da Universidade de São Paulo.
Spina G., Bartezzaghi, E., Bert, A., Cagliano, R., Draaijer, D., Bôer, H., 1996, “Strategically flexible production: the multi-focused manufacturing paradigm”, International Journal of Operations & Production Management, Vol. 16, No. 11, 00. 20-41.
Visual Object Net ++, 2003, Fakultät für Informatik und Automatisierung, Rainer Drath, <http://www.systemtechnik.tu-ilmenau.de/~drath/visual_E.htm >
Zhou, M., DiCesare, F., Desrochers, A.A., 1992, “A hybrid methodology for synthesis of Petri net models for manufacturing systems”, IEEE transactions on robotics and automation, vol. 8, no. 3, pp. 350-361.
Zhou, M. and Venkatesh, K., 1999, “Modeling, Simulation, and Control of Flexible Manufacturing Systems: A Petri Net Approach”, Intelligent control and intelligent automation, vol. 6, world scientific.

8. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.