

FAULTS AND FAULT PROPAGATION MODELING IN MANUFACTURING SYSTEMS BASED ON COLORED PETRI NETS

Miklós Gerzson¹ Adrien Leitold² and Katalin M. Hangos^{1,3}

¹Dept. of Electrical Engineering and Information System

²Department of Mathematics,
University of Pannonia
H-8201 Veszprém, P.O.B. 158,
Hungary

³Process Control Research Group

Computer and Automation Research Institute
H-1518 Budapest, P.O.B. 63
Hungary

E-mail: gerzson@almos.uni-pannon.hu

KEYWORDS

fault and fault propagation modeling, monitoring and diagnosis of manufacturing processes, colored timed Petri nets, probabilistic models, occurrence graph

ABSTRACT

A novel fault integration method is proposed in this paper for manufacturing system models given in the form of timed colored Petri nets. The faults are assumed to have a stochastic nature and are represented in the form of transitions firing in stochastic way with known fault probabilities in the system model. A novel fault effect propagation method was also developed, that can be used to compute the probabilities of the possible faulty and non-faulty intermediate and final states of the system using the probabilities of faults and the occurrence graph.

The faultless and fault containing models were implemented in CPNTools both for non-timed and timed cases. A software module was also developed for the proposed probabilistic fault propagation analysis.

The proposed methods and tools were demonstrated using a simple case study.

INTRODUCTION

Manufacturing systems form one of the traditional application fields of modeling and analysis of discrete event systems (Cassandras and Lafortune 1999). Important control related tasks, such as fault modeling, detection and diagnosis, dynamic analysis, scheduling, etc. are solved using their discrete event dynamic model in the form of Petri nets (Zhou and Venkatesh 1999). Driven by the emergence of advanced Petri net models and diagnostic tools based thereon, this field is fast developing (see e.g. (Zaytoon and Lafortune, 2013) for a recent overview).

Qualitative model based diagnostic methods – where the Petri-net based methods also belong – have been very popular in the area of fault detection and diagnosis (Venkatasubramanian et al. 2003) because they offer a systematic use of engineering knowledge about the process to be diagnosed. These methods can be classified based on the way how they describe the discrete dynamic models and their faults, and how they follow the effect of

faults during the operation of the system. Majority of the Petri net based fault detection and diagnosis methods use deterministic models, and describe the faults by hidden (unobservable) transitions (see e.g. (Dotoli et al. 2009) or Cabasino et al. 2011). However, faults often occur in practice in a stochastic way. Therefore, the aim of this work is to propose a stochastic fault modeling methodology within the Petri net approach, and to develop a fault propagation algorithm based thereon.

BASIC NOTIONS

The most important concepts are summarized here for the modeling and analysis of manufacturing systems using colored Petri nets (CP-nets) and their occurrence graph.

Timed Colored Petri Net Models

CP-nets combine the modeling advantages coming from Petri nets and compactness of the functional programming language Standard ML (Jensen et al. 2007). The CP-nets clearly enable both the mathematical and the graph representation of a manufacturing system to be modeled, where the signals of the system have discrete range space and time is also discrete (Fanti and Seatzu 2008).

Based on the formal definition of CP-nets (see details in (Jensen 1997)) the following special choices were used in modeling the faultless and faulty operation of a manufacturing process.

- **Places** refer to the operating units or their states in the modeled system.
- **Transitions** correspond to the performed manufacturing steps.
- **Tokens** are associated to work pieces and their colors are used to identify work pieces, to describe the operation to be performed on them or to describe the state of a technological element.
- **Firing time of the transitions** Some transitions are fired instantaneously, but the processing time of a manufacturing step is associated to most of the transitions.
- **Guard functions** If a manufacturing step has a fault with probabilistic nature associated to it, then its guard function is used to describe both its faulty and faultless operation together with the arc functions of the arcs adjacent to it.

Occurrence graph

The basic idea of the occurrence graph is to construct a graph which contains all of the reachable markings from a given initial one. The occurrence graph of a timed stochastic CP-net can be obtained in a similar way.

If two or more transitions or one transition with different tokens are enabled in a given state of the CP-net at the same time, then a branch appears on the occurrence graph. These firings can be in conflict or in concurrency. In our case, another reason for branching in the occurrence graph is the presence of fault during the firing of a transition when the consequence of firing, i.e. the faultless or faulty state evolves randomly.

MODELLING OF FAULT EVENTS AND INVESTIGATION OF FAULT PROPAGATION

The aim of this section is to show how the faults can be integrated into a CP-net model of a manufacturing system and how the probability of an occurring system state can be computed based on the occurrence graph of the CP-net model.

Fault Modeling

Assume that a fault or a set of faults may occur at a certain operational step during the operation of a manufacturing system. Let the occurring of these faults be mutually exclusive. Based on technological experience, let the probability of the occurring of faults is known.

The fault caused probabilistic nature of a transition t associated to a processing step can be modeled in a CP-net in such a way, that a fault function is built into the guard function of the transition t . This fault function returns the logical value *true* or *false* with predefined probability, and the token values of the adjacent consequence places of transition t can be controlled by this logical value. This type of transition firing is called a **stochastically fired transition**.

The arc expression functions of arcs starting from transition t control the consequences: either a token is put to the place representing the normal mode or it appears on one of the other places that correspond to the occurrence of a fault. The type of the occurring fault can be encoded into the color of the token. The occurring of more than two but mutually exclusive faults can be modeled in a similar way.

By using this method, faults can be integrated into the model describing the faultless operation of the system in such a way that the size and complexity of the net does not grow significantly. In our previous paper (Gerzson et al. 2012) 11 different faults were integrated into the faultless system model using this method and the resulting model was used for diagnostic investigations.

This modeling method for integration of faults can also be used when the color of tokens refer to different work pieces or to other manufacturing characteristics as it is shown in our case study below.

Fault Propagation: The Occurrence Probability of the System States

Although the faults occur randomly during the operation of a manufacturing system, the probability of their occurrence can be determined based on observed data. Having these data, it is worth investigating their effect on the probability of both the normal and the faulty system states. Here we assume, that if *more than one fault happen during the operation of the technological system, they are independent of each other*.

The occurrence graph can be used for this analysis that is generated from a given initial state of a CP-net model containing also the faulty events. *Here we assume that the occurrence graph belonging to a given initial state of a CP-net model is finite and acyclic*. The occurrence graphs of Petri nets modeling manufacturing systems fulfill this assumption in general, therefore the following considerations can be applied in a wide problem class.

First we consider the occurrence graph belonging to a given initial state of a CP-net that describes the normal, i.e. faultless operation of the modeled system. We add arc weight to the arcs of occurrence graph as follows:

1. Let the arc weight be equal to 1 if only one arc starts from a node of the occurrence graph, that is, the new system state follows from the previous one unambiguously.
2. If more than one edge start from a node, then there is a branch on the occurrence graph at this node, thus the system can get to several distinct states from the given state. The branching must have a technological reason caused by a conflict situation.

In *conflict situation* mutually exclusive firings resulting in different consequences are enabled in a given system state, and let all of their occurring probability be known and let the sum of these probabilities be equal to 1. Based on technological information the probability of each consequence can be determined, but for the general uninformed case it can also be assumed that they have the same probability. Assign these probability values as arc weights to the arcs of occurrence graph.

Next, consider the CP-net of a manufacturing system but now integrate the possible faults into the model as it was described previously. In this case new branches will appear on the occurrence graph belonging to the same initial state of the system beside the branches with technological reason. The reasons of these new branches are the occurring of faults. Assume that at the firing of an operational step t in the manufacturing system one or more faults can occur besides of normal termination. If two or more faults are possible let these be mutually

exclusive. Denote these fault events by f_1, f_2, \dots, f_k and let their probabilities be P_1, P_2, \dots, P_k , respectively. It means that while only one arc leads from the given node to the next node representing the occurring of transition t in the occurrence graph of the faultless model, at the same node $k+1$ arcs will appear in the occurrence graph of the fault containing model corresponding to the normal event (normal termination of transition t) and to the k fault events. The probability of the normal event of transition t is equal to $1-(P_1+\dots+P_k)$. Assign the probability values P_1, P_2, \dots, P_k and $1-(P_1+\dots+P_k)$ to the corresponding arcs as arc weights on the occurrence graph. The followings are true for the arc weights.

1. The arc weight is equal to 1 if only one arc starts from a node.
2. The sum of arc weights is equal to 1 if two or more arcs start from a node.

Then the *occurrence probability of the system states* can be determined as follows. Consider a node v on the occurrence graph.

1. If from node v_0 representing the initial state only one path leads to node v then the arc weights have to be multiplied along this path.
2. If there are two or more paths from node v_0 to node v then multiply the arc weights along each path and sum up these products.

These calculated values define the probability of the nodes of occurrence graph where the nodes represent the markings (token distributions) in the net, i.e. these values give the occurrence probability of system states.

In that case when the consecutive faults are not independent of each other, the conditional probability values have to be assigned to the arc of occurrence graph and the occurrence possibility of the system state can be determined using the same method.

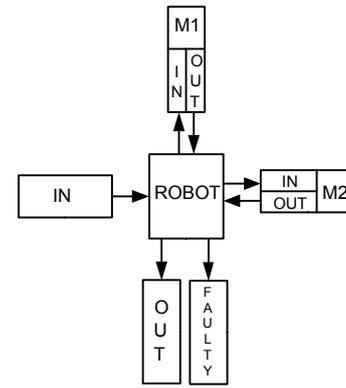
A SIMPLE CASE STUDY

The aim of this section is to illustrate the use of the proposed methods for modeling and analysis of a simple manufacturing process using CP-nets and their occurrence graph.

The software package CPNTools (CPNTools) was used for implementing the model of the manufacturing system in the form of timed CP-nets with stochastic behavior and for the occurrence graph generation. An additional software module called OGANalyzer (Leitold et al. 2013) was used to perform the probability calculations.

The Manufacturing System

The investigated simple manufacturing system contains two manufacturing lines and a robot. The scheme of the system can be seen in Fig. 1.



Figures 1: The Manufacturing System

The work pieces to be processed appear on the input bench IN. The task of the robot is to put them to the appropriate input bench of a manufacturing line M1_IN or M2_IN according to operational instructions. Assume that the two manufacturing lines perform different processing steps on the work pieces. When the manufacturing process is over, the work piece appears on the output bench of the line, i.e. either on M1_OUT or on M2_OUT. The task of robot is to put the work piece either to input bench of the other manufacturing line or to the output bench of the manufacturing line (OUT) according to operational instruction. Assume that only a single fault can occur during the manufacturing process, when the identification label of the work piece can get damaged. Then work pieces with unreadable label get into a separate container (FAULTY). A detailed description of the system operation can be found in (Leitold et al. 2013).

The Petri net of the described manufacturing system consist of the following elements. Places and tokens on them give the number and type of work pieces on input and output benches. Another place refers to the status of the robot: if there is a token on it then the robot is available. Places describe the state of input and output benches of lines, and token on places refer to the fact that the transfer process is under way. A transition refers to the manufacturing processes, and further transitions refer to the start and the end of transfer processes. The color of tokens contains a work piece identifier and the code of manufacturing process(es) to be carried out. The guard function assigned to a transition of a manufacturing step generates the fault randomly with a user defined probability. In case of timed simulation of the manufacturing system different time units are assigned to the transitions appearing as transition inscriptions on the net. A part of the CPN model of the system can be seen in Figure 2.

Analysis based on the occurrence graph

For the illustration of the generation and analysis of the occurrence graph, let us have the following marking after the generation of work pieces:

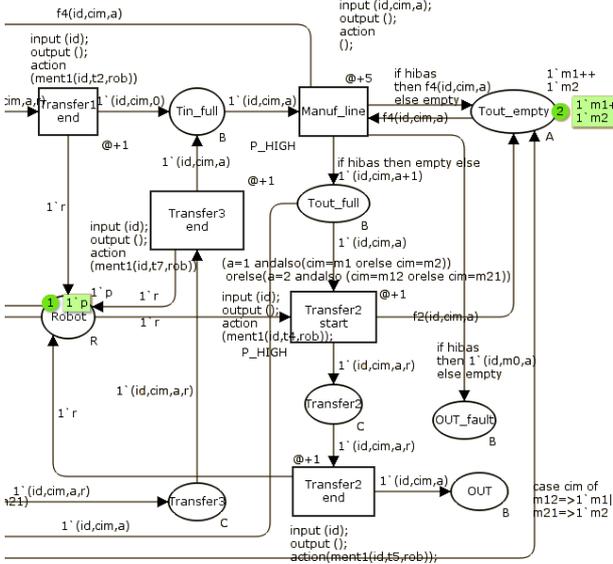


Figure 2: Part of CPN Model of a Manufacturing System

- There are two tokens ($1'(1,m2)$, $1'(2,m12)$) on place IN representing two work pieces to be processed the first on Line 2, the second first on Line 1, then on Line 2;
- There is a token ($1'p$) on place Robot denoting the robot is ready to work;
- The tokens ($1'm1++$ $1'm2$) on input (Tin_empty) and output places (Tout_empty) of manufacturing lines indicate the emptiness of input and output places.

This token distribution is used as initial marking for the investigations of CP-net model both in normal and in faulty modes. We investigated the model with non-timed and timed transitions. The detailed description of the investigations and the resulted occurrence graphs can be found in (Leitold et al. 2013). It is important to note that the occurrence graph is acyclic as a consequence of the identification of work pieces.

Because of the small number of work pieces, the resulted occurrence graphs are relatively simple, and the analysis

of nodes can be done manually. Thus it is easy to find the terminal nodes referring to the normal faultless termination of the process, and those terminal nodes where the manufacturing of either of the pieces or of both of them ends with fault. If the number of pieces becomes larger the size of the occurrence graph grows exponentially.

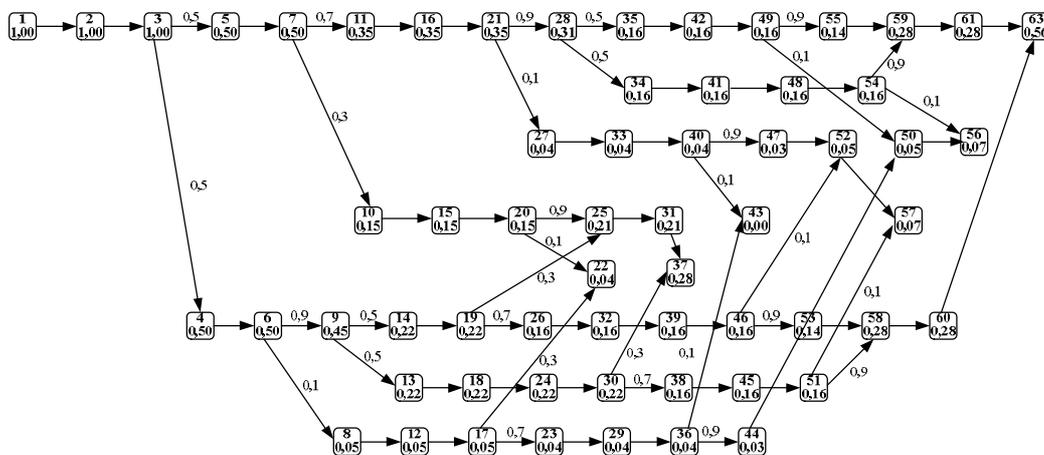
Probabilistic Analysis of Occurrence Graph using the OGANalyzer

Unfortunately, CPNTools cannot use the information about the probability of faults at the generation of occurrence graph. However, assigning this value to the appropriate edges of the occurrence graph, the probability of each node on the occurrence graph, i.e. of each system state can be determined. To calculate the probabilities, a software module called **OGAnalyzer** has been developed (Leitold et al. 2013).

Non-timed case. In the **non-timed** case there are three branches on the occurrence graph (see in Fig. 3.) with technological reason (nodes 3., 9. and 28.) because the robot can choose randomly between the transferring of the two work pieces. The value of arc weights assigned to arcs starting from these nodes are 0.5 - 0.5. The reason of all the other branches is the possible occurrence of faults. The arc weights of the other branches are calculated by the pre-defined fault probabilities. In case of manufacturing line 1, the arc weight belonging to fault occurring is equal to 0.3, while in case of normal operational mode it is 0.7. In case of line 2 these values are 0.1 (faulty case) and 0.9 (normal case). There are 6 terminal nodes on the occurrence graph.

The probability of nodes, that is the **probability of system states**, can also be calculated using the above described method.

Let *node 57* be considered. This node refers to the situation when the manufacturing of work piece labeled by m2 is completed without fault but the label of the other



Figures 3: The Reachability Graph of Non-timed Case in OGANalyzer

work piece m12 gets damaged during the processing on line 2 and this work piece gets into the waste product container.

There are three different routes through the branches leading to node 57 caused by technological reasons. Multiplying the arc weights along these routes the summing these products the probability of this node is equal to 0.07. The probability of the normal termination of the manufacturing process (node 62), i.e. the manufacturing of both work pieces completed without any fault can be calculated in a similar way and it is equal to 0.56.

The timed case. Using the timing values i.e. in **timed case**, the occurrence graph becomes a tree with 12 terminal nodes. These 12 nodes refer to 6 different process terminations and the difference between the pairs is processing time. Because of timing, only one branch has technological reason and the leaving arcs have the same probability. All the other branches are caused by faults, and the arc weights can be determined similarly to the non-timed case. Because of the tree structure there is only one route to all nodes from the root, and the probability of nodes can be calculated by multiplying the arc weights along the routes.

CONCLUSION

A novel occurrence graph analysis procedure for discrete event systems described by timed colored Petri nets was proposed in this paper for model-based diagnostic purposes. The faults are assumed to have a stochastic nature and were represented in the form of stochastic transitions with known fault probabilities in the system model that became a timed colored Petri net with stochastic behavior. The color of tokens representing the work pieces were used to distinguish them and to assign a label of the processes to be carried out. The arc inscriptions and the built-in probability function were used for the fault integration.

A novel fault effect propagation method was also developed, that can be used to compute the probabilities of the possible faulty and non-faulty final states of the system using the probabilities of faults and the occurrence graph. This graph was also used for the behavioral analysis of the model. A special software module, called OGAnalyzer was also developed for the handling of the probabilities on the occurrence graph and for calculating the occurrence possibility of system states.

The proposed methods and tools were illustrated using a simple case study.

ACKNOWLEDGEMENT

This research is partially supported by the Hungarian Research Fund through grant No. K-83440. We also

acknowledge the financial support of the Hungarian State and the European Union under the TAMOP-4.2.2.A- 11/1/KONV-2012-0072.

REFERENCES

- Cabasino, M.P.; A. Giua; M. Pocci and C. Seatzu. 2011. "Discrete event diagnosis using labeled Petri nets. An application to manufacturing systems" *Control Engineering Practice* Vol. 19. 989–1001.
- Cassandras, C.G. and S. Lafortune. 1999. "Introduction to Discrete Event Systems" Kluwer Academic Publishers
- Dotoli M; M. P. Fanti; A. M. Mangini and W. Ukovich. 2009. "On-line fault detection in discrete event systems by Petri nets and integer linear programming" *Automatica* Vol. 45. 2665-2672
- Fanti, M.P. and C. Seatzu. 2008. "Fault diagnosis and identification of discrete event systems using Petri nets" *9th International Workshop on Discrete Event Systems WODES 2008*, 432-435
- Gerzson, M.; B. Márczi and A. Leitold. 2012. "Diagnosis of Technological Systems based on their Coloured Petri Net Model" ARGESIM Report Eds: Troch, I., Breitenecker, F. no. S38 358/1-6.
- Jensen, K. 1997. "Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use" Springer-Verlag
- Jensen, K.; L.M. Kristensen and L. Wells. 2007. "Coloured Petri Nets and CPN Tools for Modelling and Validation of Concurrent Systems" *Int. J. of Software Tools for Technology Transfer* No. 9. Vol. 3-4 213-254
- Leitold A.; B. Márczi; A.I. Pózna and M. Gerzson. 2013. "Investigation of manufacturing systems using timed colored Petri nets" *Factory Automation Conference, Veszprém, Hungary*,
- Murata T. 1989. "Petri Nets: Properties, Analysis and Applications" *Proceedings of the IEEE*, Vol. 4. No. 9
- Yaqiong L.; C.K.M. Lee; H.K. Chan and W.H. Ip. 2012. "RFID-based colored Petri net applied for quality monitoring in manufacturing system" *The International Journal of Advanced Manufacturing Technology* Vol. 60. 225-236
- Venkatasubramanian V.; R. Rengaswamy and S. N. Kavuri. 2003. "A review of process fault detection and diagnosis Part II: Qualitative models and search strategies" *Computers and Chemical Engineering* Vol. 27. 313-326
- Zaytoon J. and S. Lafortune. 2013. "Overview of fault diagnosis methods for Discrete Event Systems" *Annual Reviews in Control* Vol. 37. 308–320
- Zhou M. and K. Venkatesh. 1999. "Modeling, Simulation, and Control of Flexible Manufacturing Systems: A Petri Net Approach" World Scientific, Singapore

WEBREFERENCES

- CPNTools 2.2.0 <http://wiki.daimi.au.dk/cpntools/>, University of Aarhus, Denmark, CPN Group