IV. Fuzzy Petri Nets and Their Application in CIME*

Robert Valette, David Andreu

LAAS-CNRS

Janette Cardoso,

LCMI-EEL-UFSC,

Jean-Claude Pascal

LAAS-CNRS

Key words

Petri nets, Fuzzy Petri nets, Manufacturing Systems, Uncertainty and Change

1. Preliminary discussion**1.1 Why Fuzzy Petri nets are useful for CIME?**

Factory automation has been addressed at two very different levels in the past. Either the organization of the production system, the planning of the workflow, the scheduling of the operations are improved but the elementary operations are not necessarily automated. Or the major part of the effort is dedicated to the automation of the elementary repetitive sequences of operations by means of Programmable Logic Controllers the management of the production remaining largely based on human heuristic decisions.

The first approach deals with an overall model of the production process. What has to be improved and automated is a complex decision making system where human is present. The data are frequently imprecise; ill-known information has to be taken into account, the environment is poorly structured and unforeseeable (unexpected events are frequent). Not surprisingly, all available Artificial Intelligence techniques are typically used in this approach, including fuzzy sets and possibility theory⁽⁸⁾⁽²⁶⁾.

In the second approach, the environment is generally well-known, human is not included in the system to be automated. It is possible to formally specify the control/command sequences by means of Petri nets and to automatically derive programs for Programmable Logic Controllers.

The long term trend is towards integrated automation and concurrent engineering which requires to associate and combine these two approaches. It is one of the motivations for so much work concerning Petri nets and

logic and more specifically Fuzzy Petri nets
(1)(3)(8)(12)(21)(23)(24)

1.2 Historical review

Fuzzy Petri nets can be considered as resulting from attempts in combining logic and net theory. M. D. Zisman⁽²⁷⁾ has perhaps been the first one to point out that Petri net transitions could be interpreted as rules in specific production rule systems. Another milestone is the seminal work by T. Murata and D. Zhang proposing a Petri net modeling of logic programs⁽¹³⁾. In the meanwhile, fuzzy set theory⁽²⁵⁾ was used to elaborate a theory of possibility⁽²⁶⁾ and the concept of fuzzy rule has been the basis of fuzzy control⁽²²⁾, a prominent technological breakthrough.

As a Petri net transition could be interpreted as a rule and as the concept of fuzzy rule has been so rich, it is a very natural approach to try to understand what can be the result of combining Petri nets and fuzzy sets. In the Petri net literature, the ondest paper defining a fuzzy Petri net is the one of C. G. Looney⁽¹¹⁾. Then various models have been elaborated⁽³⁾⁽⁵⁾⁽⁹⁾⁽¹⁵⁾⁽¹⁹⁾⁽²³⁾. Before detailing and comparing in section 2 the various approaches which have been developed, let us discuss in more detail where, in the field of factory automation, Fuzzy Petri nets are likely to be useful.

1.3 Where applying Fuzzy Petri nets in CIME?

Petri net is a formal tool for describing a *Discrete Event System* model of an actual system i. e. its representation by means of a set of states and a set of state changes (events). With respect to state machines (or regular languages), the advantage of Petri nets is that concurrent evolutions (various processes evolving simultaneously and partially independently) can be easily represented and analyzed. In local control applications, conditions/events nets (safe i.e. 1-bounded Petri nets) are used to describe the *control sequences* of elementary devices. At the level of supervisory control (or overall

* This work has been partially done in the framework of PRC-GDRIA, the French Research Programme for Artificial Intelligence, sponsored by the Ministry for Research and by CNRS.

performance evaluation of a manufacturing system), High-Level nets (Petri nets where data structures are attached to tokens) are used to represent the *flow of parts* in the shop.

Fuzzy sets⁽²⁵⁾ are a way of building an interface between two views: a *symbolic* one (for instance, x is an element of set X , or x has a property denoted by X) and a *gradual numeric* one (for instance, the membership function μ_x characterizing X takes the value 0.83 for x , $\mu_x(x)=0.83$). In the context of automation, two completely different way of employing fuzzy sets have been developed.

In the first approach (fuzzy hontrol⁽²²⁾), there is no ill-known information. The control policy is defined by means of a set of rules defining what action is to be done according to the situation of the controlled system. As the set of rules is finite, the set of the situations taken into consideration is also finite. When the physical systems are continuous, fuzzy sets are used to build *linear interpolations* between the considered situations. The symbolic view (set of rules) provides an easy, clear understanding of the control policy to the human operator.

In the second approach⁽⁸⁾, fuzzy sets are used to refine a decision process. When a set of constraints is so tight that no solution is possible, the some constraints have to be relaxed (it is the role of a negotiation when the management is uniquely done by humans). In order to choose which constraints have to be relaxed, it is necessary to introduce a *preference relation*. The fuzzy set memberships functions are used to denote this preference as well as to deal with ill-known values and dates⁽⁶⁾⁽⁷⁾⁽²⁶⁾.

Within CIME, Fuzzy Petri nets are promiseful when they are likely to be a cross-fertilization of typical Petri net applications and typical fuzzy set theory ones. Let us enumerate some examples:

a) **Local control**: Very often, Programmable Logic Controllers are supplemented by simple regulation functions. However these functions cannot be called in the body of a sequential control program and no formal model authorizes the description of a continuous control leading a system from one state to another one within a sequence description. This can be achieved by considering that the firing of a transition is a continuous process: the tokens progressively disappear from the input places and appear in the output places. It is the basic idea of *fuzzy firings* in a safe Petri net⁽¹⁶⁾.

b) **Supervisory control**: Various applications can be imagined at this level. The first one concerns diagno-

sis. Typically, diagnosis involves ill-known information. The propositions which have to be handled are hypotheses (abductive reasoning). It is a natural application of possibility theory⁽⁶⁾ which is based on fuzzy sets in order to attach a grade to the hypotheses which are more or less likely to be true. Typically, monitoring manufacturing systems is based on inconsistency analysis between the events generated by the system behavior and those generated by the Petri net model describing its normal operation. It is therefore natural to introduce reasoning involving ill-known states and sequences in such a context⁽¹⁾⁽³⁾⁽⁴⁾⁽⁵⁾. Indeed, if the model is too strict, too precise, frequent meaningless deviations between the behavior of the system and that of the model are to be expected. If it is too loose, too imprecise, then it is of no use. Fuzzy sets are a convenient tool to establish a trade-off between these two situations. In continuous industrial processes threshold values are "fuzzified", in discrete event systems, events have also to be "fuzzified" which leads to another application of Fuzzy Petri nets⁽²³⁾.

c) **Management and human/computer interface**:

A possible application concerns the modeling of the dynamic behavior of a collection of actors (humans and computers) which are cooperating in order to manage a manufacturing system. The organization dimension of the firm is considered (it is a concurrent engineering approach) and the interaction between the planning decision center, the shop loading one, the scheduling one etc, is analyzed. It is a typical example of so called Cooperative Computer Supported Works (CCSW). It is generally recognized that Petri net could be useful in this field but they are restricted to the representation of standard, normal behaviors in order to avoid the combinatorial explosion resulting from the description of all the exceptions. However, human requires a high degree of autonomy which cannot be captured by the kind of strict specification which are obtained by means of Petri nets. Introducing imprecision and illknown information by means of fuzzy Petri nets could be a solution which has yet to be explored.

2. How combining fuzzy sets and Petri nets?

2.1 What can be fuzzified in a Petri net?

Petri nets are made up of places, transitions and tokens. A state is represented by a distribution of tokens in the places (marking). Firing a sequence of transitions transforms the state and corresponds to a sequence of events. One of the major results in Petri net theory is that p-invariants and t-invariants can be

computed by linear programming *without enumeration of all the reachable markings*. A p-invariant is a set of places for which the overall token load remains constant whatever transitions are fired. A t-invariant is a sequence of events that can repetitively occur i.e. such that the system is cyclically passing through the same states.

Various approaches can be used to combine Petri nets and fuzzy sets. It is possible to define a first kind of fuzzy markings by denoting the place token loads by means of a fuzzy number. It is also possible to define fuzzy markings by attaching to each token its fuzzy location characterized by a fuzzy set of places (the places where it is likely to be located). A third alternative is to associate fuzzy variables with the tokens i.e. to use the data structure attached to the tokens in high level nets. Finally, it is possible to define fuzzy firing sequences which are more or less likely to be fired.

The combination of Petri nets and fuzzy sets will only be promising if it is consistent. This means that something of the Petri net theory has to be respected. The evolutions of the fuzzy markings by means of transition firings have to be consistent with the p-invariants and t-invariants of the underlying ordinary Petri net.

Let us consider the Petri net in figure 1. a. This net has two positive p-invariants: $\{p_1, p_2, p_4\}$ and $\{p_1, p_3, p_5\}$. The initial marking is such that place p_1 contains 3 tokens ($M_0(p_1)=3$) and the other ones are empty. Let us assume that the system evolves during a certain time and that the current marking is ill-known because we do not exactly know how many times transitions t_a, t_b, t_c and t_d have been fired. The token loads $M(p_i)$ are ill-known but, from the two p-invariants, we are absolutely certain that:

$$M(p_1) + M(p_2) + M(p_4) = 3$$

$$\text{and } M(p_1) + M(p_3) + M(p_5) = 3$$

Dually, if we consider a firing sequence including t_a and t_d , the firing order of t_b and t_c may be unknown but we are certain that both of them have been fired after t_a and before t_d .

In consequence it is essential to develop approaches

Fig. 1 Petri net with initial marking

which are both capable of capturing ill-known information and consistent with the well-known Petri net structure. Let us now briefly present some approaches which have recently been developed.

2.2 Fuzzy Petri nets as models for reasoning

This approach has been clearly presented in⁽⁵⁾. References⁽⁹⁾⁽¹¹⁾⁽¹⁵⁾⁽¹⁹⁾ have strong relationships with it. The Petri nets are implicitly considered as Condition/Event nets (safe nets) and the authors concentrate on the firing sequences. They are concerned with the modeling of reasoning and thus each transition firing denotes the application of an inference rule. When a token is put in a place, this means that the corresponding proposition is proved true. Typically, in a reasoning process, given a proposition you never prove its truth twice. It is therefore natural to consider that the nets are acyclic and that no sequence results in putting two tokens in a place.

Nevertheless, some authors⁽¹⁹⁾ have considered this situation which results in some inconsistencies with Petri net theory. Actually, in this approach, when a place receives a second token it is merged with the first one because it is considered to be the composition of two different proofs having the same conclusion. In a way it is assumed that once marked, a place is never emptied until a new proof scenario is considered.

In⁽⁵⁾ this assumption is almost explicit (the authors introduce the notion of reachability set) and it is clear that the main purpose is to compute a degree of "likelihood" for each firing sequence leading from an initial marking to a final marking. With each token a degree of truth is attached and with each transition a degree of likelihood. When a transition is fired, its output place are marked with tokens which have truth degrees equal to the minimum of the truth degrees of the input tokens multiplied by the likelihood degree of the transition.

In diagnosis problems, it is essential to compare possible hypotheses which could explain the observed abnormal situation. If the abduction rules are represented by fuzzy Petri net transitions and the propositions which are more or less likely to be true, by places, the firing sequences are possible explanations. The obtained Petri nets are acyclic and only t-invariants are meaningful because they denote complete firing sequences leading from the hypotheses (source transitions) to the observations (sink transitions). The Petri net in figure 1. b is covered by two t-invariants: $\{t_e; (t_f \parallel t_h)\}$ and $\{t_e; (t_g \parallel t_h)\}$ meaning that after firing t_e either t_f and t_h (unordered) are fired or (exclusive or) t_g and t_h (unordered). If this net is the ordinary underlying net of a fuzzy Petri net, the sequences describing more or less

likely scenarios (inferences) derived from the fuzzy Petri net have to be consistent with these t-invariants. A scenario which only fires t_e and t_f would be incomplete because the side-effect t_h is missing.

As possible approaches dealing with diagnosis issues or with inconsistent knowledge with a solid Petri net background the papers⁽¹⁴⁾⁽¹⁸⁾ have to be cited. They do not explicitly deal with fuzzy Petri nets but the relationships with the above papers are strong. In particular, the work by T. Murata and V. S. Subrahmanian⁽¹⁴⁾ could be a starting point of an approach in which the fuzzy information is stored as a token attribute in a high level net.

2.3 Fuzzy Petri nets as models for physical system

In this context, a Petri net does not denote a reasoning process. It is a model of a physical system, either at a low level of abstraction for the local control, or at the level of the overall shop for the supervisory control. The places describe the various states of physical objects or devices. They are typically covered by p-invariants because an object (denoted by a token) is always in one and only one state. Actually, given an object, the collection of places describing its states is a p-invariant. For instance in the net in figure 1.a the p-invariant $\{p_1, p_2, p_4\}$ could be interpreted as the set of states of a transport device (automated guided vehicles) and $\{p_1, p_3, p_5\}$ that of the parts in some manufacturing system. The common place p_1 would denote a transport during which a part is on an automated guided vehicle. At the initial marking we have three pairs of objects consisting of a part on a vehicle.

Let us consider the case of the local control. The Petri net is typically a safe net and, in this case place p_1 only contains one token (one part and one vehicle). The token denoting the vehicle moves along the invariant $I_1 = \{p_1, p_2, p_4\}$ and the one denoting the part along $I_2 = \{p_1, p_3, p_5\}$. A fuzzy marking, as defined in⁽¹⁶⁾⁽¹⁷⁾, associates with each invariant a fuzzy set of places which are the possible locations of its unique token. Let us assume that L_1 is the fuzzy location of the vehicle in I_1 and $\mu_{L_1}(p_2) = 0.5$ and $\mu_{L_1}(p_4) = 1$. This means that the current situation matches two characteristic discrete situations: one with the vehicle in p_4 and partially (grade 0.5) one with the vehicle in p_2 . A linear interpolation can be made in order to compute a command consistent with this intermediary state. In⁽¹⁶⁾⁽¹⁷⁾ this is used to implement control rules such as "slow down from state p_2 to state p_4 " by attaching the output command "goes ahead" to p_2 and "stops" to p_4 and by making the token gradually pass from p_2 to p_4 .

At supervisory control, the Petri nets describing manufacturing systems are not safe but, on the other hand, the tokens are individuals i.e. they carry information and are all different⁽²⁰⁾. Instead of attaching a fuzzy set of places to each p-invariant, it can directly be associated with the tokens⁽³⁾⁽²³⁾. For instance, let us assume that there are three vehicles v_1, v_2 and v_3 , the fuzzy marking of invariant I_1 will be defined by three fuzzy sets of places $L_1(v_1), L_1(v_2)$ and $L_1(v_3)$. When diagnosis is involved, we indeed deal with ill-known and incomplete information. Consequently, if we have $\mu_{L_1(v_1)}(p_2) = 0.5$ and $\mu_{L_1(v_1)}(p_4) = 1$ this means that p_4 and p_2 are possible locations (p_4 is the more likely) for v_1 . However, from the marking definition, we are certain that v_1 (as well as v_2 and v_3) is somewhere among p_1, p_2 and p_4 because for any reachable marking M of the underlying Petri net we have $M(p_1) + M(p_2) + M(p_4) = 3$. In⁽²³⁾ this notion of fuzzy marking is complemented by a temporal reasoning. It is used to verify that every event occurrence is consistent with the previsual fuzzy date computed by the management decision level (planning and scheduling).

3. Conclusion

We have presented some fuzzy Petri net approaches and their possible applications in the area of manufacturing systems. According to the fact that the net is used to depict a reasoning scenario or the physical part of a system, the important part of Petri net theory with which the fuzzy Petri net approach has to be consistent is either characterized by its t-invariants or by its p-invariants. In the first case, the approach has to focus on uncertain sequences; in the second case on imprecise markings.

A unified framework covering these two kinds of fuzzy Petri nets has yet to be developed. The opinion of the authors is that linear logic could be one element of this⁽¹⁰⁾⁽⁴⁾ because this logic differs from the ordinary first order logic by the fact that the propositions are resources which are produced and consumed and this is also the main feature which differentiates Petri nets from *And/Or* graphs.

(Manuscript received May. 26, 1994)

References

- (1) T. Cao & A. C. Sanderson: "Sensor-based Error recovery for robotic task sequences using Fuzzy Petri nets", 1992 IEEE International Conference on Robotics and Automation Nice France, (May 1992)
- (2) T. Cao & A. C. Sanderson: "Variable reasoning and analysis about uncertainty with fuzzy Petri nets, Lecture Notes in Computeyr Science" *Application and Theory of Petri nets*

- 1993, 691, Ed. M. Ajmone Marsan, p. 126~145 (1993)
- (3) J. Cardoso, R. Valette, & D. Dubois: "Petri nets with uncertain markings, in Lecture Notes in Computer Science" *Advances in Petri nets 1990*, 483, Ed. G. Rozenborg, Springer Verlag, p. 64~78 (1991)
 - (4) J. Cardoso, B. Pradin Chézalviel, & R. Valette: "Fuzzy Petri Nets and Linear Logic", *IEEE/SMC Int. Conf. on Systems, Man and Cybernetics: Systems Engineering in the Service of Humans*, Le Touquet, (Oct 17~20 1993)
 - (5) Shyi-Ming Chen, Jyh-Sheng Ke, & Jin-Fu Chang: Knowledge representation using fuzzy Petri nets, *IEEE Trans. on Knowledge and Data Engineering*, 2, N° 3, p. 311~319, (1990)
 - (6) D. Dubois & H. Prade: "Possibility Theory: an approach to computerized processing of uncertainty", Plenum Press, (1987)
 - (7) D. Dubois & H. Prade: Processing fuzzy temporal knowledge, *IEEE Trans. on Systems, Man and Cybernetics*, 14, N° 4, p. 729~744, (1989)
 - (8) D. Dubois, J.C. Pascal, & R. Valette: "Fuzzy decision and control in manufacturing and automation", Chapter 47 of the *Handbook on Manufacturing and Automation*, R. C. Dorf and A. Kusiak Ed., J. Wiley and Sons Inc., to appear in 1994.
 - (9) M. L. Garg, S. I. Ahson, & P. V. Gupta: "A fuzzy Petri net for knowledge representation and reasoning" *Information Processing Letters* 39 p. 165~171 (1991)
 - (10) J. Y. Girard: Linear Logic; *Theoretical Computer Science*, 50, (1987)
 - (11) C. G. Looney: Fuzzy Petri nets for rule-based Decion making *IEEE Trans. on Systems Man and Cybernetics* 18, N1, p 178~183 (Jan-Feb 1988)
 - (12) J. Martínez, P. R. Muro, M. Silva, S. F. Smith, & J. L. Villarroel: "Merging artificial intelligence techniques and Petri nets for real-time scheduling and control of production systems", *12th IMACS World Congress on Scientific Computation*, Paris, , 3, p. 528~531 (July 1988)
 - (13) T. Murata & D. Zhang: "A Predicate-Transition net model for parallel interpretation of logic programs", *IEEE Transaction on Software Engineering*, 14, 1, (April 1988)
 - (14) T. Murata, V. S. Subrahmanian, & T. Wakayama: "A Petri net model for reasoning in the presence of inconsistency", *IEEE Trans. on Knowledge and Data Engineering*, 3, 3, p. 281~292 (Sept. 1991)
 - (15) V. Olej, J. Chmúrny, & I. Mokris: "Analysis of paralld discrete systems by discrete stochastic and fuzzy Petri nets", *Computer and Artificial Intelligence*, 10, 3, p. 221~237 (1991)
 - (16) J.C. Pascal & R. Valette: "A Petri net based fuzzy PLC for linear interpolation between control steps IPMU'92 International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems", Palma de Mallorca, Spain, July 6~10, 1992 also in *Uncertainty in Intelligent Systems*, North Holland, p. 297~305 (1993).
 - (17) J.C. Pascal, R. Valette, & D. Andreu: Fuzzy sequential control based on Petri nets ETFA'92 IEEE International Workshop on Emerging Technologies and Factory Automation, Melbourne, Australia, p. 140~145 (August 11-14, 1992)
 - (18) L. Portinale: Exploiting t-invariant analysis in diagnostic reasoning on a Petri net model, *Application and Theory of Petri Nets 1993 Lecture Notes in Computer Science* 691, Springer Verlag, p. 339~356 (1993)
 - (19) H. Scarpelli & F. Gomide: "Fuzzy Reasoning and High Level Fuzzy Petri Nets", *First European Congress on Fuzzy and Intelligent Technologies*, Aachen, Germany, (Sep 7-10 1993)
 - (20) C. Sibertin-Blanc: High-level Petri nets with data structures, 6th European Workshop on Application and Theory of Petri nets, Helsinki, Finland, (June 1985)
 - (21) M. Silva & R. Valette: Petri nets and Flexible Manufacturing, in *Advances in Petri ness 1989*, Lecture Notes in Computer Science 424, Springer Verlag, p. 374~417 (1990)
 - (22) M. Sugeno: An introduction survey of fuzzy control, *Information Science*, 36, (1985)
 - (23) R. Valette, J. Cardoso, & D. Dubois: "Monitoring Manufacturing Systems by means of Petri Nets with Imprecise Markings", *IEEE International Symposium on Intelligent Control 1989*, Albany N. Y., USA. (sept. 1989)
 - (24) R. Valette & M. Courvoisier: "Petri nets and Artificial Intelligence, *IEEE/SICE International Workshop an Emerging*

Technologies for Factory Automation, Palm Cove, Cairns, North Queensland, Australia, p. 218~238 (August 17-19 1992)

- (25) L. A. Zadeh: Fuzzy sets, *Information and Control*, N° 8, p. 338~353, (1965)
- (26) L. A. Zadeh: "Fuzzy sets as a basis for a theory of possibility", *Fuzzy sets and systems*, N° 1, p. 3~28, (1978)
- (27) M. D. Zisman: Use of production systems for modeling asynchronous concurrent processes, in *Pattern Directed Inference systems* D. A. Watterman and F. Hayes-Roth (Eds.), Academic Press, London, p. 53~68 (1978).

Rebert Valette received his Ph. D. Degree from the University "Paul Sabatier" of Toulouse (France) in 1976. He is currently Senior Researcher at LAAS-CNRS, Toulouse France. His current research interests include the application of Petri net theory in manufacturing systems and the study of the relationships between Petri nets and logic.

D. Andreu received the M. S. degree in Computer Control from the Paul Sabatier University in 1992 and he is currently a Ph. D. student at LAAS-CNRS. In his research works, he has developed a fuzzy programmable logic controller and he is presently working on the control of batch processes (hybrid systems).

Janette Cardoso received her M. Sc. Degree from the Federal University of Santa Catarina (UFSC) of Florianópolis (Brazil) and then her Ph. D. Degree from the University "Paul Sabatier" of Toulouse (France) in 1990. She is currently Professor at UFSC and doing her research work at the LCMI. Her current major research interest is the development of a consistent approach combining Petri nets and possibility logic.

J. C. Pascal received the Ph. D. degree in Control Science from the Paul Sabatier University of Toulouse in 1987. From 1987 to 1989, he worked in Aeronautical Industry as software design engineer. Since 1989, he is an Assistant Professor of Computer Control Engineering and Control Science at Paul Sabatier University. His research interests, at LAAS-CNRS, are on discrete-events systems control using Petri nets and Fuzzy theory and turn to discrete-events/continuous systems control such as batch processes.