A Colaborative Fuzzy CPN System for Conflict Solution of Flexible Manufacturing System

Felipe B. Reis, Vinicius F. Caridá, O. Morandin Jr., Renan L. Castro, Carlos C.M. Tuma
Department of Computer Science
Federal University of São Carlos (UFSCar)
São Carlos, Brazil
{felipe_reis, vinicius_carida, orides, renan_cafe, carlos_tuma}@dc.ufscar.br

***Abstract*—The flexible manufacturing systems (FMS) is composed of machines, robots and automated transportation systems. Make the control of an FMS is a complex task due to the many subsystems and elements that compose it. Due to the large number of variables to be analyzed, the modeling and control of these systems become complex tasks, difficult to maintain and adapt to different layouts. The purpose of this paper is to indicate a strategy for modeling FMS based on Petri nets (PN) allowing the control to be performed by an external system. Thereby the modeling is easy to understand, requires low efforts to adapt to different production plans, and allows control of the FMS model take into consideration information current state of the system in decision-making for conflict resolution. To support control system is built multi criteria fuzzy rules. *CPN Tools* will be used for modeling, simulating and Matlab will be used for creating the fuzzy system.**

***Keywords*—manufacturing modeling, fuzzy system, colored Petri nets, conflict.**

I. INTRODUCTION

In the current context of the economy, manufacturing industries are facing a crucial challenge: to increase the productive capacity of their systems to answer the volatile demands and production cycles getting shorter. Flexible Manufacturing Systems (FMS) have been designed to attend market needs that demand products that have low cost, quality, and prompt delivery. The complexity of FMS control increases with the needs of production and technological resources that cannot be designed using sequences of discrete events, simple and / or interlocks basic. The FMS must meet the needs of the production plan, alternative routes of production, operating system and the availability of resources, beyond basic control [4].

The control problem is related to the characterization of the system as being from a discrete event. The purpose of a controller for these systems is to establish and maintaining a sequence of events desired. The system must be free of deadlocks and the controller should not create new situations like this. The controller must also satisfy a set of restrictions [5].

System modeling and its representation have been considered as one of the most important challenges in many fields of engineering. A deep analysis of the manufacturing system model is an essential step before construction of a real system [1].

Colored Petri Nets (CPN's) have been used as a promising technique for modeling automated manufacturing environments. There is software that can generate graphical models of CPN's. Some offer the possibility of applying simulation on the generated models and even make mathematical analysis. Many works have been developed using CPN modeling, such as [6], [7], [19] ensuring their efficiency.

Petri nets can efficiently model a manufacturing system with conflicts, buffer size, precedence relationships, resource sharing and structural interactions [2]. When two or more processes share a resource in a manufacturing system, this may result in a conflict situation.

Conflicts modeled in a Petri net belong to free choice net class [3], where two or more arcs leaving the same input place are destined to different transitions. This generates different execution paths and only one transition among the enabled ones can be fired at a certain instant of time. Uncertainty about the continuation of the transition firing requires a decision.

In turn, the automatic control plays a vital role in the advance of engineering and science and has become an integral and important industrial processes and manufacturing [8]. A technique of artificial intelligence widely used for decision-making systems is fuzzy logic.

This paper proposes the modeling of a manufacturing system using CPN and fuzzy control. The production routing, AGVs routes, machines and buffers are modeled in CPN and a rule-based system for decision making is built using Matlab. It was developed a system which has the purpose communicate the modeling with the control, i.e., communicate the fuzzy system with Petri net and combining the efficiency of fuzzy systems for decision making and modeling in Petri nets. Using this approach it is possible to settle a dynamic scheduling, control the system and ensure that the system is caught when unexpected events occur, such as machine breakdowns or changes in production management, because only real time information are used based on the system state to solve the conflicts.

II. Conflicts in Manufacturing Systems

Several studies have been proposed to model manufacturing systems and resolve conflicts when resources are shared. The performance of an FMS is highly dependent on the efficient allocation of the limited resources and it is strongly affected by the effective choice of scheduling rules [9].

Dispatching rules have been widely used to resolve conflicts in the input queue to use a shared resource [10]. Rules like shortest processing time (SPT), earliest due date (EDD), and first come first served (FCFS) have been used.

An approach in [11] is proposed to model and scheduling a manufacturing system using Petri nets and dispatching rules in order to resolve any conflicts. In [12] a manufacturing system model based on agents is represented by Petri nets. The agent based system continuously monitors the status and conditions of the system and makes decisions related to production.

The authors of [13], [14] and [15] used simulation to find an efficient rule to assign priorities. Therefore, considerable time is spent on simulation.

In these approaches based on dispatching rules and agents, conflicts are solved considering single criteria. Some studies have used fuzzy logic to build dispatching rules combining several criteria to be met.

The proposal of [16] presents a fuzzy system for production scheduling in an FMS. A simulation model of FMS and a fuzzy system is constructed and at various points of the system, decision points are considered to verify their status and decision making.

In [17] is proposed the use of fuzzy logic to choose the next step among the possible routing alternatives for the processing of a product.

The proposal of [7] was presented a modeling manufacturing system using Fuzzy Colored Petri nets that resolves conflicts system considering system information in real time. The characteristic of use system variables to solve conflicts in real time covering one of the disadvantages of the research [19]. The paper [7] propose an approach to control in an intelligent manner, considering system information such as, for example, the quantity of products in buffer. A disadvantage of the work is the increase in complexity of understanding or reading of the model. Another disadvantage is the difficulty of adapting this model to other production plans and finally the configuration of this model in real systems requires processing equipment control high capacity.

III. Proposal

The purpose of this paper is to provide an efficient model for solving conflicts in a Flexible Manufacturing System (FMS). A FMS is a complex system due to various subsystems and elements that composes it, alternative production routing and concurrent events. Consequently, conflicts can occur between processes.

Static scheduling approaches generated at the production planning usually solve the problem, but they can easily become infeasible in a real manufacturing environment since they assume highly unrealistic assumptions [18].

A dynamic scheduling was chosen because it considers the current state of the shop floor at specified points and variables as buffer's actual space and AGV location can be used to make a decision.

Conflicts as resource competition and alternative production routing are solved using fuzzy logic. The system is modeled in an extension of Petri nets, the Colored Petri Net (CPN) and when there is a conflict an external system of fuzzy decision is invoked.

In the proposed model, system state information at the time of decision is considered. Therefore, it is possible to ensure its stability. If there are changes in production management, machine breakdowns or unavailability of material, the production planning will not have to be redone, since the decision, from the time of the occurrence of unanticipated events, consider the current state of the system. The alternatives to be considered in the conflict solution are only those that are feasible to run.

One of the advantages of perform the fuzzy reasoning using a fuzzy system outside modeling it in Petri nets, as a facility to modify and add input or output variables, change the rule base, the inference and defuzzification method, and also facilities to develop in a object-oriented language.

Analysis methods, like cover ability tree, also can be used. These properties allow a complete analysis of the model to detect deadlock, ensure the absence of non-reachable states and verification of place limits.

In the CPN model is possible to model the dynamic system, its resources and properties. The fuzzy system enables real-time decision and resolution of conflicts. The two separate communication systems is through a web service of communication ensuring Performed efficiently.

Some criteria will be used to solve processes conflicts. Information as total processing times, buffers level, queue time, remaining processing time, due date and lot size will be considered. Fuzzy sets for these variables were defined and production rules were constructed by an expert. After the evaluation of the rules, priorities are given to all process in conflicts. Then an order is determinate for manufacturing.

Decisions begin in the production orders. In the load station is decided which product should be made first. The variables used in this solution are the lot size and due date. Fuzzy sets and fuzzy rules are built for this decision.

If the product has two or more alternative routing to be processed, one routing is chosen considering the processing time in the routing and the buffers level of the machines in this routing. After one routing is select, the part will be processed in it until is finished. This is done to all products at the load station.

If the next machine is idle, the processing starts as soon as the machine arrives at the input buffer. Besides, it stays in the buffer until the machine is not busy anymore. In the meantime, if arrive other parts at the buffer, two criteria are used, remaining processing time and queue time, to choose the part that will be processed first.

When a part finishes a processing step, it goes to the output buffer. If there is more than one part in the buffer, one part is chosen based on the same criteria used in the input buffer selection.

An AGV is selected to transport the part to the next machine. If there is no more steps to go, then the product goes to the unload station

IV. MODELING

To model the proposal, this paper uses a modeling in *CPN Tools* of a complex manufacturing system proposed by [19]. In [19], a modeling strategy for control and interlocking of an Automated Manufacturing System using Virtual Petri Nets (VPN) was built. A supervisory system is simulated to make decision and solve conflicts at the load station, input buffer, or whenever an AGV is free.

A *top-down* technique was used to model the system. The system is composed of six machines with load/unload areas and input/output buffers and three AGVs. A load/unload station and AGV park are also considered in the model.

In this work, *CPN Tools* were used to model manufacturing systems and modeling the CPN for decision making. *CPN Tools* also enables the verification and validation of the model and can be used to investigate Petri net properties.

The modeling of FMS is made considering the physical aspects of FMS in accordance with the paper [19], wherein the modeling process is divided such that it can be in the form of steps:

- In the first step is considered the layout for initial elaboration of the model.
- The second step is concerned with the start of interlocking transport system. With relation to transport, should also, start the placement of inscriptions on the arches to assign routes to products. Also at this step must consider the definition of types to be assigned to places.
- In the third step are considered some products or transport vehicles. They are required to be able to run the first simulation, which considers the flow of these elements by the system.
- In the fourth step is the modeling of the workstation and their buffers.
- In the fifth step are added to the remaining inscriptions on the arches with the aim of define the complete flow of products.
- In the sixth step is performed the task of connection between modules.
- The seventh step is reserved specifically for simulation.
- Finally, the eighth step adds time to the model.

Thus, modeling should represent the workstation in the system, their areas of input buffer and output buffer of products. AGVs, AGV parking area, warehouse of parts and products, as well as the path in which occur the flow of AGVs between areas of workstation buffers, parking and storage.

In modeling the places represent the workstation, buffer areas of input and output parts, AGVs parks and warehouse of parts and products. The path or flows of AGV are represented by arcs. The limits of parts in buffer, amount of AGVs in particular workstation, number of AGVs in the parking and amount of AGVs in input buffer or output parts, can be established by making use of logical places within the network model or the external control system. In this paper the limits were established through logical places that regulate this limit. The names of these limiters places were prefixed with inhib_. They ensure for example that only one AGV is the front of a workstation or node, every moment. Fig. 1 shows the model after the first steps described above.

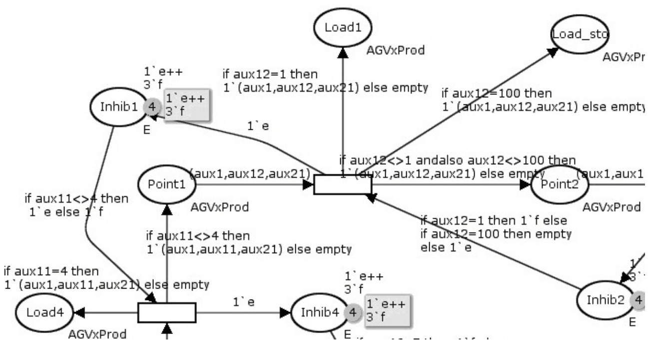

Figure 1. Exemple of the firt steps of modeling

A Interface Model

After the modeling done representing the physical aspects of the system and limits the presence of AGVs, pieces and products in the machines, buffers, parking and warehouse, you must insert control places in points to possibility of conflict.

For each place that represents a conflict situation the transition that precedes it must be removed. In Figure 2 are shown the transition that will be removed and the arches dependent or related to the transition.

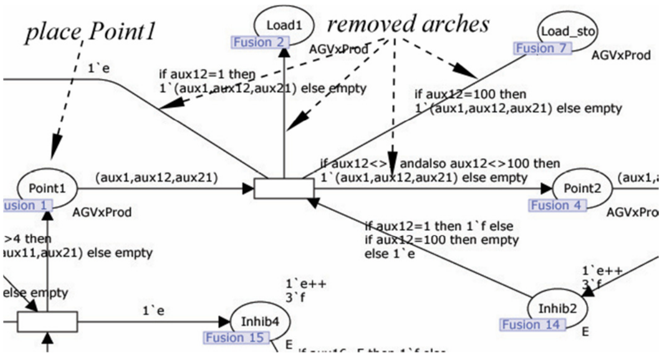

Figure 2. Exemple of the firt process of the interface model

Then we must add a transition between each place that precedes the removed transition. Each of these transitions added should receive an place. This place is called the control place. As illustrated in Fig. 3.

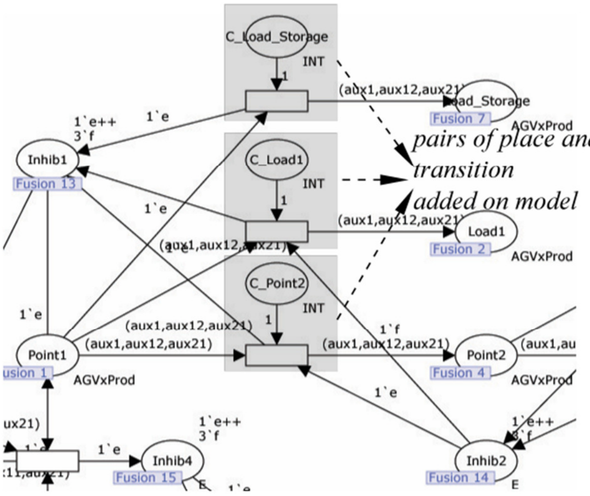

Figure 3. Exemple of the second process of the interface model

This process should be repeated at all points or places where there is a conflict, making the network dead at the same time creating a control interface for that external systems may interfere with the performance of the network while it is at runtime.

B Interface Control System

The control system interface is the control channel used for communication between the control system and the Petri nets model.

The framework Access / CPN is a library of code written in Java that allows direct communication with models written in CPNTools. This code library has been extended to a web service that offers the services of change of mark in places, recovery marks in places, and execution of transitions enabled.

To extend the library were created classes and methods responsible for integration, such as:

- Simulator class that owns the method Start () responsible for establishing communication with the CPNTools.
- PNWatcher class that represents the model in Petri nets. It is class through the member Simulator communicates with CPNTools and then communicates with the Petri nets of the system model.
- PlaceWatcher class has 3 methods: getMark (String placename), setmark (placename String, String mark) and execute ();
- GetMark method receives as parameter the name of a place of the petri nets and returns a string containing the value or token of the place at the time of the call.
- Setmark method receives as parameter the name of a place and a token. This token is inserted in place passed as parameter.
- Execute() method has the role of firing all transitions enabled at any given time of the implementation of Petri net.

C Control Framework

To allow scalability of the control system has created a framework of classes that allow all physical elements of the modeled FMS have a high-level representation and so a schedule easy to maintain and requires less effort to adapt the models to other FMS systems.

The Framework has classes that deal with the state reading of the model. Thus it is possible that a token on the model is read and interpreted.

The definitions of the types or colors in CPNTools allow this extraction is done. Therefore there must be a communication protocol between the model and framework so that this last can interpret the information contained in the token. For example, the state representation of an AGV is in the form of a tuple: (AGV, INT, PROD), wherein:

- Agv, represents the name of AGV;
- Int, represents the fate that the AGV must follow;
- Prod, represents the type of product being transported by AGV.

This representation occurs in the model and the framework is responsible for reading these values to be transmitted in high level to the control system.

D Control System

The conflict resolution of the model is realized by a decision framework based on fuzzy. A fuzzy system was built using the Matlab tool. The function of the fuzzy system is receiving the information from the control system when it is in conflict situations and considers this information in order to decide which will best way to resolve the conflict.

The fuzzy system returns to the control system the decision and control system is responsible for reflect this decision in the model inserting tokens in control places giving condition for the transitions are triggered and the system continues to run of the model.

The control interface was constructed to lock the model whenever a conflict situation is reached. In this state the model alone is unable to continue running unless a token is inserted in control places.

The control system through the control framework decides at runtime how the control should be done or what conflict resolution should be taken and the active control places giving a condition for the transition to be triggered and execution of the model be continued. Therefore the control system is divided into two main parts: the control, and decision making.

The control is also divided into two parts: parking control, and AGVs control.

The parking control is responsible for verifying the model and find out if in system there are demand for transport. If there is demand for transport the control system inserts a token in place of the control parking and releases a AGV that this demand is met.

TABLE I. RELATIONSHIP OF CONFLICT SYSTEM AND POSSIBLE DECISION MAKING

Status / Position of AGV	**Conflict situation**	**Variables Considered**
Free / Point junction (node)	Choose output buffer; Choose product; Choose input buffer;	Number of products in the output buffers; Products in the output buffers; Number of nodes; Distance to travel; Quantity of products in the input buffers;
Free / Heading for the output buffer	Entering the buffers area; Proceed to the next node;	Destination of the AGV Current Node
Busy / Point junction	Entering the buffers area; Proceed to the next node;	Destination of the AGV Current Node
Busy / Addressing to the input buffer	Entering the buffers area; Proceed to the next node;	Destination of the AGV Current Node

The AGVs control is responsible for determining which transportation demand will be met. Before the AGV meets a demand for transport the AGV control activates the fuzzy system through decision framework and receives information which transport demand should be met by AGV.

Decision making is performed whenever there is a conflict situation in the system. In Table I is illustrated the potential conflict situations in the system.

The control triggers decision making whenever the system enters a conflict state. With this composition the control system ensures that the model will run with all conflicts situations resolved. Figure 4 presents the main module of the proposed model.

V. CONCLUSIONS

The challenge of this work was evolving in issues of modeling and control of FMSs.

This paper suggests an approach to model a flexible manufacturing system with shared resources considering conflict points to make decisions and perform an execution. One of the aspects of this proposal is that the decisions consider real time information based on the system state.

The improvement over the proposal of [19] was to increase the representativeness of the model even with the assurance that the system is free of conflicts. Compared the proposal of [7] progress is by reducing the complexity of understanding or reading model. Another advantage is the ease of adapting this model to other production plans and configuration of this model in real systems requires processing equipment for control of low capacity. The model allows a control system to work due to decrease makespan and the idleness of the resources of the model of FMS.

The final model integrates all sub models and then steps of simulation were performed. This modeling can be used to realize a production scheduling in real time and control the manufacturing systems.

REFERENCES

[1] H. Lee; A. Banerjee "Modeling and Representation of Manufacturing Process with incertainties Using Parametric Modular Design and xPNML", International Symposium on Collaborative Technologies and Systems, 2009.

[2] A. A. Desrochers; R. Y. Al-Jaar, "Applications of Petri Net in Manufacturing Systems - Modeling, Control, and Performance Analysis", IEEE Press, New York, 1994.

[3] B. Hrúz, M. C. Zhou, "Modeling and control of discrete-event dynamical systems: with Petri nets and other tool", Advanced textbooks in control and signal processing, Springer, 2007.

[4] O. J. MORANDIN, E. R. R. KATO, "Virtual Petri nets as a modular modeling method for planning and control tasks of FMS". Systems, Man and Cybernetics,. p. 1521 - 1527 vol.2. , 2003.

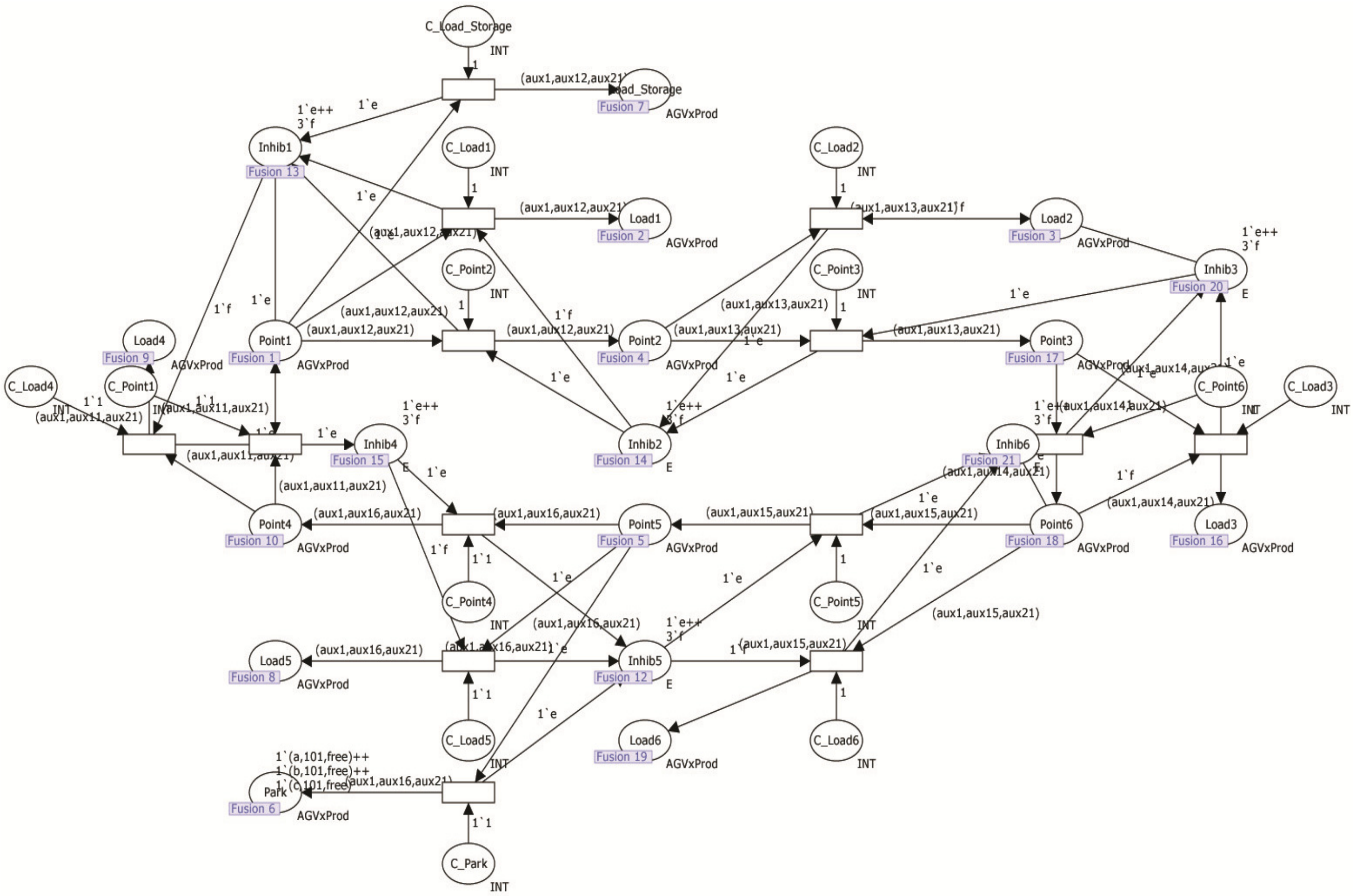

Figure 4. Main module of the model proposed

[5] A. A. DESROCHERS; R. Y. AL-JAAR, "Applications of Petri Nets in Manufacturing Systems: Modeling, Control, and Performance Analysis". 1995.

[6] L. GOMES, "On conflict resolution in Petri nets models through model structuring and composition". IEEE International Conference on Industrial Informatics, p. 489 – 494, 2005.

[7] E. R. R. Kato, O. Morandin, M. Sgavioli, "A conflict solution manufacturing system modeling using Fuzzy Coloured Petri net," Systems Man and Cybernetics (SMC), pp.3983-3988, 2010.

[8] K. OGATA, "Modern control engineering". 2. ed. Rio de Janeiro: Prentice-Hall do Brasil, 1992.

[9] G. A. Tuncel, "Heuristic Rule-Based Approach for Dynamic Scheduling of Flexible Manufacturing Systems", Multiprocessor Scheduling, Theory and Applications, 2007.

[10] F. T. S. Chan; H. K. Chan; H. C. Lau, "The State of the Art in Simulation Study on FMS Scheduling: A Comprehensive Survey. International Journal of Advanced Manufacturing Technology", 19:830–849, 2002.

[11] G. M. Delgadillo; S. P. Llano, "Scheduling Application using Petri Nets: A Case Study: Intergráficas S.A.", 19th International Conference on Production Research, 2007.

[12] S. Turgay, "Agent-based FMS control. Robotics and Computer-Integrated Manufacturing", vol. 25, 470-480, 2009.

[13] K. K. Lee, "Fuzzy rule generation for adaptive scheduling in a dynamic manufacturing environment", Applied Soft Computing, v.8 n.4, p.1295-1304, 2008.

[14] H. Zhang; Z. Jiang; C. Guo, "Simulation Based Real-time Scheduling Method for Dispatching and Rework Control of Semiconductor Manufacturing System", IEEE International Conference on Systems, Man and Cybernetics, 2901-2905, 2007.

[15] B. Shnits; D. Sinreich, "Controlling flexible manufacturing systems based on a dynamic selection of an appropriate operational criteria and scheduling policy", International Journal of Flexible Manufacturing Systems, v18, 2006.

[16] F. T. S. Chan; H. K. Chan; A. Kazerooni, "Real Time Fuzzy Scheduling Rules in FMS. Journal of Intelligent Manufacturing", 14,341-350, 2003.

[17] I. Bilge; M. Firat; E. Albey, "A Parametric Fuzzy Logic Approach To Dynamic Part Routing Under Full Routing Flexibility", Computers And Industrial Engineering, 15-33, 2008.

[18] W. Xiang, H.P. Lee, "Ant colony intelligence in multi-agent dynamic manufacturing scheduling", Engineering Applications of Artificial Intelligence,21,73–85, 2008.

[19] Araújo, R. G., "A Modeling Method For Ams Control Using Virtual Petri Nets". São Carlos, April 2006.

[20] M. C. Fernandes; A. H. de Freitas; Junior, O.M.; E.R.R. Kato, "Production re-sequencing for manufacturing based on discrete event simulation and fuzzy inference", IEEE International Conference on Systems, Man and Cybernetics, 2003.