

Modeling and Analysis of an On-Line FMS Scheduler Using Colored Petri Nets

Ali A. Al-Titnchi

Electrical & Computer Eng. Dept,
Applied Science University, Jordan.
email:aaali8@yahoo.com

Kasim M. Al-Aubidy

Computer Eng. Dept,
Philadelphia University, Jordan.
email:kma@philadelphia.edu.jo

Abstract: *An interactive hierarchical model based on colored Petri net (CPN) for general flexible manufacturing system (FMS) scheduler is presented in this paper. The proposed model is partitioned into two levels to face the complexity of the manufacturing systems. The first level (cell-level) supervises the jobs scheduling between cells. The second level (machine-level) supervises the scheduling of the manufacturing jobs between machines inside the active cells. The goal of the scheduler model is to interpret the output of the manufacturing route planner into graphical representation and to achieve the sequencing, dispatching and monitoring the real time execution of these manufacturing plans. Also, the proposed CPN model generates feedback status to monitor the abnormal conditions. Sub CPN priority structures are embedded in the main model to control the election from the competitive jobs to use the limited resources in the system. The hierarchy organization will enhance the real time response of the sequencing and monitoring capabilities. The designed scheduler has been evaluated for different case studies and the achieved results are very encouraging.*

Keywords: *Colored Petri nets, On-line scheduling, Real-time computing, Discret event systems, Distributed systems, Flexible manufacturing systems.*

Received: January 2004 | **Revised:** March 29, 2005 | **Accepted:** April, 25, 2005

1. Introduction

A flexible manufacturing system is a real-time production system that can be quickly configured to produce multiple types of products or jobs [15]. In this case, a discrete number of raw parts are processed and assembled by computer-controlled machines. Flexible manufacturing systems consist of numerically controlled machine centers, industrial robots, automated vehicles, material handling, ...etc. In a highly automated FMS, products are processed according to a job scheduling. This scheduler is a dynamical discrete event system whose events include: loading, processing, unloading, occurrence of machine failures, repairing of machine tools, etc. In the case of a sudden change of production plans or a failure of machine tools or transportation links, it is necessary to obtain a new schedule immediately. Therefore, on-line scheduling has become very important task for obtaining efficiency and high productivity [16].

Complex and real-time FMSs are hard to model and analyze. Several approaches [11], [12], [18] have been

proposed to cope with FMS modeling requirements. Among those, Petri nets have been widely used due to their well known features [4], [9]. The major advantage of using Petri nets is the evaluation of the system status before implementation. Nevertheless building Petri net models of real-time FMSs is not an easy job, since the configuration of FMS may change over the system lifetime and during normal operation.

The idea of using Petri nets for the modeling and analysis of distributed systems is not new. Hatono and Tamura [6] describe a rule-based on-line scheduling system and an FMS simulation system under certainty using stochastic Petri nets, which developed for creating, debugging, and evaluating the rule-base for on-line scheduling. However, simple Petri nets models tend to become highly complex for such systems. In this case, high-level Petri nets, called Colored Petri Nets are recommended [5], [7], [8], [15]. The approach to modeling concurrent and real-time systems with colored Petri nets is described by Pettit and Gomaa [13]. For designing manufacturing systems, Colombo and Carelli[4] introduced an approach based on CPNs. A

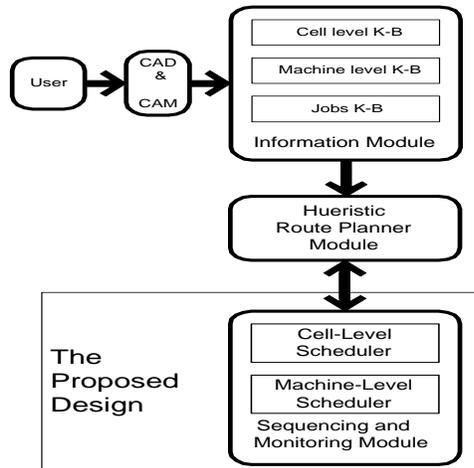


Figure 1. System organization.

genetic algorithm is used to simultaneously find the near-optimal resource allocation and the event-driven schedule of a closed Petri net. The resulting Petri nets are then compared with the Petri nets optimized for a particular production plan to address the effectiveness of the optimization [15]. This hybrid scheme is not guaranteed to be optimal. Another model methodology based on CPN for representing manufacturing systems can be used to support the dynamic nature of the system operation including rescheduling and failure recovery [14]. Chen and Chen[3] developed an object-oriented approach based on CPN, to modeling of FMS dynamic tool allocation and control under a non-hierarchical shop floor control scheme. The proposed method can provide the designer of a tool management system with a high-level and structured representation of the tool-sharing control.

The concepts of routes and CPN were proposed for modeling the supervisor for flexible manufacturing systems [2]. Such a model can be very complex when using place/transition nets. To overcome this problem, CPNs can be used in obtaining a more compact model with the same desired behavior. A hierarchical route planner for FMS's based on heuristic algorithm were introduced by Al-Titinchi and Al-Aubidy [1].

This paper deals with the design and simulation of an interactive on-line scheduler using CPNs. The paper is organized in seven sections. Section 2 outlines the interaction between the manufacturing route planner and the scheduling stages. The design and operations of the cell-level and machine-level scheduler CPN models are described in sections 3 and 4 respectively. The scheduler priority algorithms used in this work are given in section 5. Section 6 outlines a typical case study and the parameters used in the system evaluation. Finally, section 7 concludes this paper.

2. The Scheduler Organization

In our previous work [1], the design and implementation of a hierarchical route planner for FMS's based on heuristic algorithm were introduced. The aim of the planner is to obtain the optimal manufacturing routes for jobs according to well-designed cost function. This paper deals with an extension of our previous work, as illustrated in Figure 1. The sequencing and monitoring module is an on-line scheduler modeled using CPN rules. It consists of two levels, the cell-level and the machine-level. The cell-level monitors the competitive jobs to use the manufacturing cells. The machine-level monitors the competitive jobs to use the machines inside the manufacturing cells of the first level. Also, a feedback is proposed in the model to discover the abnormal conditions in the system such as machine breakdown, and transportation link damage. This feedback which activated through the use of sensors are added to the CPN model as extended arcs. These arcs will pass signals to the route planner to modify the old

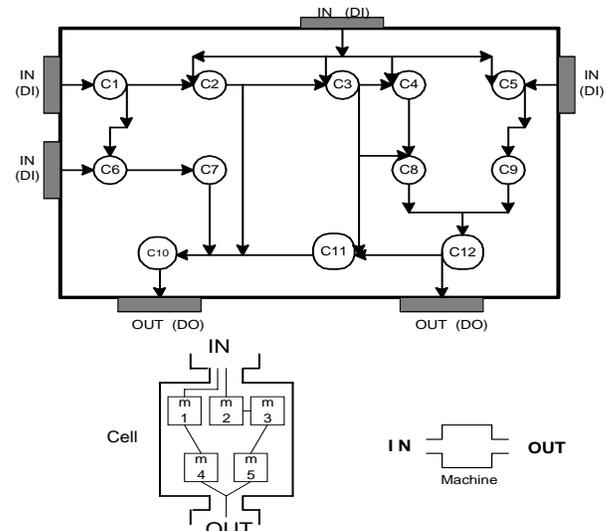


Figure 2. General layout of an FMS

manufacturing routes to avoid problems that may occur.

It is obvious that CPN is a compact modeling tool for description of automated manufacturing system. Moreover, the CPN rules are suitable to achieve the monitoring of real time activities of interaction between the jobs to be manufactured, and the physical components, which are; manufacturing cells or machines, transportation links, storage buffers, and local robots. In fact, two factors affect the monitoring of the real-time job's flow in the FMS environment, these are; the manufacturing routes resulted from the route planner, and the CPN models represent the scheduling tasks.

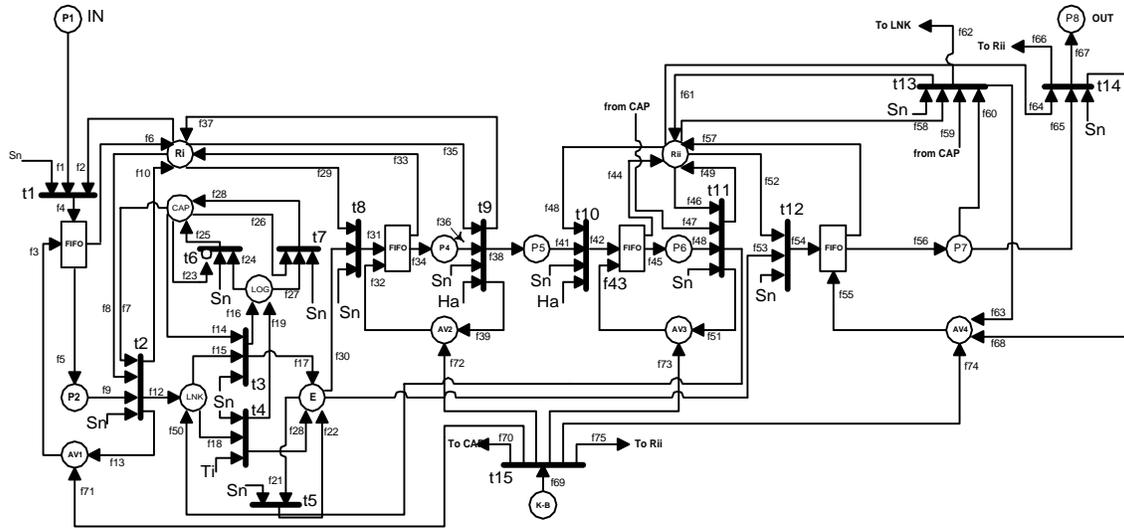


Figure 3. The CPN model for cell-level scheduling.

The proposed scheduler will have the following features;

- The CPN models work with multi cell/multi job manufacturing system illustrated in Figure 2.
- The scheduler design has a hierarchical scheme incorporating two main parts; the cell-level and the machine-level.
- The aim of the suggested scheduler is to monitor and control the concurrency, synchronization and priorities between the jobs to use the limited manufacturing resources.
- Four CPN models are recognized in the scheduler design, these models are;
 - The cell level scheduler model.
 - The machine level scheduler model.
 - The FIFO algorithm model.
 - The heuristic priority algorithm model.

Appendix (A) describes the definitions of all places (p), and transitions (t) used in this paper.

3. Cell-Level Scheduler

The scheduler of the FMS at cell level has been designed using the concepts of CPN, as illustrated in Figure 3. It will control the sequencing and dispatching of the competed jobs to use the manufacturing cells. Three parameters will cooperate to make the decision about the jobs flow within the model. These parameters are;

- The manufacturing route for each job.
- The FIFO algorithm.
- The colors set and the functions that label the arcs of the cell level model itself.

The scheduler activity starts when jobs enter the FMS through the place p_1 , which represents such group

of jobs in dummy input buffers. The transition t_1 will fire when $Jx1$ comes through f_1 , and there is workable robot through f_2 . However, t_1 can not fire unless the sensors indicate no problem in the resources through Sn . On firing t_1 , the token $\langle Jx1, Ry \rangle$ will enter the FIFO model. Now, on the availability of the input buffer $[INz]$ across f_3 , an output of FIFO will be fed to P_2 through f_3 as token $\langle Jx1, INz, N \rangle$ while the related robot will be released through f_6 . The argument N means the priority values given to the related job by the FIFO policy model. It is important to clarify that different colored tokens mean different jobs and resources. When (f_7, f_8, f_9, Sn) are valid according to their definitions, t_2 will fire and;

- release the related robot and a location in the input buffer across f_{10} and f_{13} respectively,
- load the transportation link through f_{12} according to the result of the route planner.

Now the transportation mechanism assumes that the transportation will take place either across t_3 (when the transportation link is fallen, i.e. when the place (CAP) becomes empty or across t_4 after a certain period of time (T_i).

After firing either t_3 or t_4 , the jobs will move to the place (E) where tokens are of the shape $\langle Jxs, Lew, N \rangle$. This token means such a job on the related transportation link with such a priority value. From E, t_5 will fire which causes a movement of one displacement by the related link. Otherwise, t_8 or t_{12} will fire with respect to any job that reach such a target on it's manufacturing route. However, t_8 and t_{12} represent deferent ways. The transition t_8 takes the jobs toward the input buffer of such a cell, while t_{12} takes the jobs to an intermediate or external output buffers.

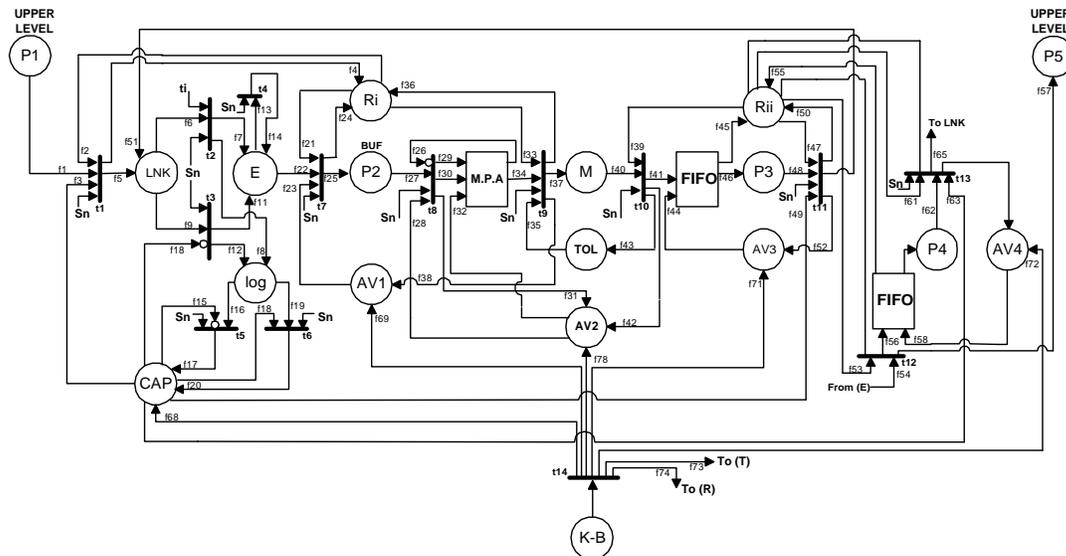


Figure 4. The CPN model for machine-level scheduling.

However, which of them will fire at any stage for each job depending upon the resulted manufacturing routes from the route planner. The next part of the transportation mechanism consists of the place (LOG) and the two transitions t_6 and t_7 . The place (LOG) works as logical representation for the locations already become free on the initial position of the related links in the FMS. Hence, its color tokens are of the shape $\langle Le0 \rangle$, where 0 means the starting position of a transportation link. The $\Sigma \langle Le0 \rangle$ in (LOG) will excite t_6 and t_7 depending on the function f_{23} . If f_{23} indicates that (CAP) is empty then t_6 will fire, otherwise t_7 will fire. As explained, any transportation link will end either at t_8 or t_{12} . For instance, when a transportation link reaches the entrance of such a cell, then the jobs on that link which are decided by the route planner will be delivered to the cell by the local robot across f_{29} . When t_8 fires, this will take jobs to inside the FIFO policy model, and then push these jobs after tagging them with priority values to the input buffer of the related cell. The transition t_9 will fire if it is enabled by (f_{35} , f_{36}) and the activation signals from S_n and H_a . The tokens $\langle Jxs, Ca, N \rangle$ in P_5 means the jobs (Jxs) in a cell (Ca) where each job with its priority (N). The activity detail inside P_5 is the responsibility of the machine level CPN model. When job's manufacturing process is completed inside the cell, the signal H_a for this cell will be active at t_{10} . At this stage the jobs will travel through f_{42} , FIFO and f_{45} to the related output buffer in P_6 . The jobs in the same place P_6 don't mean the same physical buffers. This really depends upon the argument [Ba2] of the token $\langle Jxs, Ba2, N \rangle$ in that place. From P_6 the jobs will be guided according to their manufacturing routes through f_{50} to the required transportation link.

Again, at this point the manufacturing route will inform the model either to schedule the job to another cell through t_8 as described above or to translate the job across f_{53} to enable t_{12} . If t_{12} fires two types of tokens may flow through f_{54} to P_7 . These are $\langle Jxs, Ob, N \rangle$ or $\langle Jxs, Df, N \rangle$. The first type means that the manufacturing of the job is completed and it is in an external output buffer (OUT), ready to get out of the FMS through f_{67} . The second type is understood as the job in such an intermediate buffer (Df), and it is ready to be handled by the transportation links system through the sequence f_{60}, t_{13} , and f_{62} . To complete the explanation of the model, two more points must be clarified. The first is to reduce the model complexity by representing the local robots availability by two places (R_i) and (R_{ii}). The second is the transition (t_{15}) will feed the model from the knowledge base place (K-B). Its firing will decide the initial conditions of the buffers and transportation links capacities in order to avoid the overload of these resources.

4. Machine-level scheduler

Figure 4 shows the CPN model of the machine level scheduler. It supervises the following activities:

- the manufacturing route of the jobs between the machines inside the cells
- the competition of the jobs to use the shared resources in the cells
- the abnormal conditions or damaging in the resources.

In other words, this model is really a zooming for the detailed activities inside place (P_5) of the cell level.

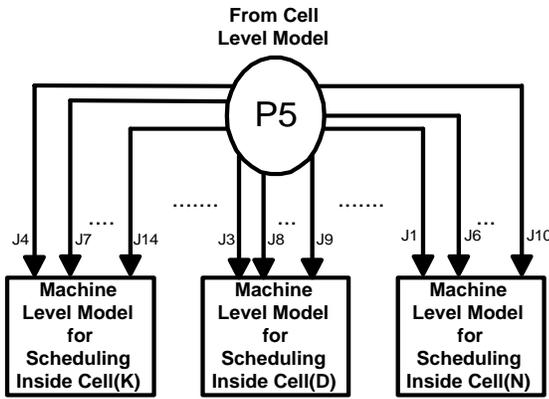


Figure 5. The relationship between cell-level and machine-level CPN models

Figure 5 shows an example for the relation between the cell-level and the machine-level. However, the physical resources at this level, which interact with the jobs to be manufactured, are machines, transportation links, storage buffers, and robots.

As in the cell level model, three factors achieve decision making at this level, these are:

- the manufacturing route for each job that resulted from the route planner,
- the basic rules that define the places, transitions and the arcs functions of the CPN model itself, and
- the proposed priority policies which solve the competition problem to use the resources. These policies are the FIFO and the heuristic priority algorithms.

The machine-level model can be described in the same manner of the cell level model. Only some differences must be taken into account, these are:

- The colored tokens should be defined to deal with the components of the machine level. These are the machines, machines buffers, local robots,...etc.
- The functions (Sn) represent the sensors, which indicate the hardware condition and validation of the active elements inside a cell.
- The places (P2) and (P3) are equivalent to (P4) and (P6) in the cell level model.
- The place (m) is the manufacturing location in this model. It represents jobs manufactured at machines. This place is equivalent theoretically to the place (P5) in the upper level, which considers the cells as the manufacturing locations.

5. The Scheduler Priority Algorithms

Two CPN priority algorithms are considered; the FIFO algorithm, and the heuristic priority algorithm. The FIFO model is used to control the sequencing of the jobs when they arrive to utilize the transportation system.

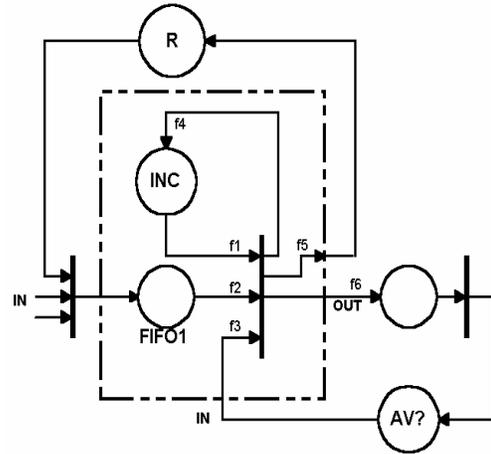


Figure 6. The CPN model for FIFO priority.

Hence, this model supports both the cell and the machine levels. The heuristic model is designed to achieve the sequencing of the jobs when they arrive to utilize the machines at machine level.

5.1. The FIFO model

This model works according to the principle of first in first served policy. Its operation starts at the arrows labeled IN, as illustrated in Figure 6. At these points the model is connected to the cell-level and machine-level models. Assume that a job is coming into this model, it will wait at the place FIFO1 until the availability of the required resource through f3 is valid. At this moment the transition can fire and hence the job will be tagged with current priority value. Which is coming from the place INC. The job with it's given priority will pass across f6 to the system. These token will be either in the form $\langle Jxs,?,N \rangle$ for cell level or $\langle Jxt,?,N \rangle$ for machine level. It is important to clarify that (?) in the token means which resource such as $(Inz, Ba1, \dots)$. Moreover, on firing the transition, function f4 will increment the value of the priority in place [INC] for the related resource. However, this model is of theoretical aspect. In other words no physical movement is achieved through the activities inside this model. Hence, the token in the place FIFO1 does not mean that a robot transfers a job to this place. In fact, it means only logical movement to label the job with the current priority value.

5.2. The heuristic priority model

Figure 7 illustrates the heuristic priority CPN model. Six priority policies, used as sequencing rules, are incorporated into this model. These rules are heuristically informed from the knowledge base already produced by the route planner. These rules are:

- Rule 1: Select the job with the highest user priority.

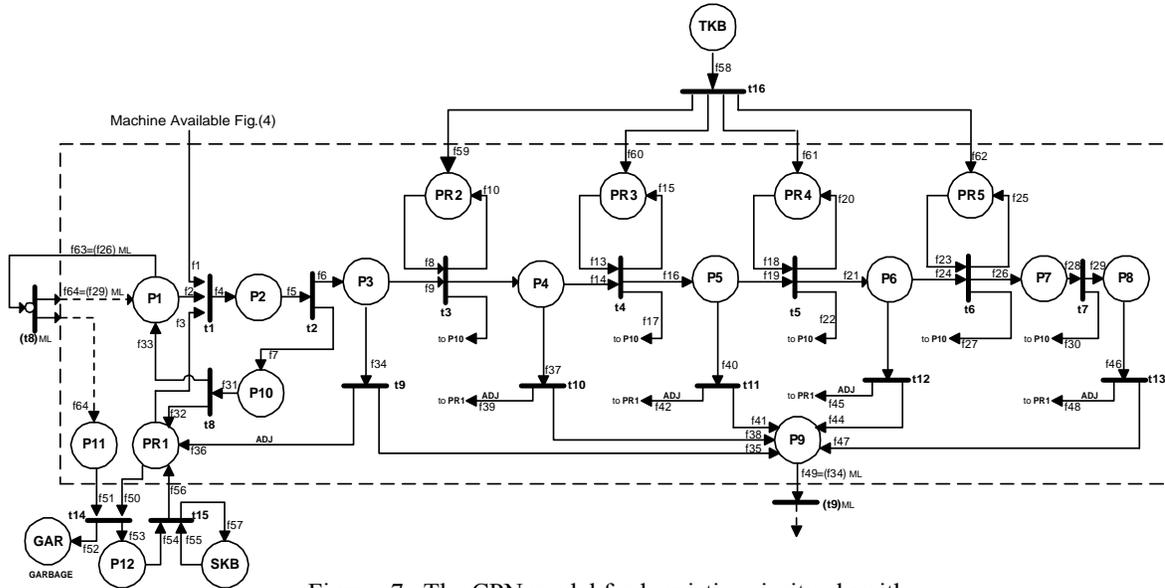


Figure 7. The CPN model for heuristic priority algorithm.

- Rule 2: Select the job with minimum processing time on the current machine.
- Rule 3: Select the job with minimum processing time on the previous machines.
- Rule 4: Select the job with the longest processing time on the next machines.
- Rule 5: Select the job with highest number of operations on the current machine.
- Rule 6: Select the job arbitrarily.

If more than one job is selected by a priority rule, the next rules will perform further selection until one job is decided to be processed on the machine before the others. However, the importance of the priority rules in the model is not fixed. It depends upon the expertise's requirements and this will achieve the generality and flexibility of the design. Also, the model can be extended for any number of priority rules if other parameters are required to control the sequencing of the jobs at the shared machines. Figure 4 shows the relation between the machine level model and this heuristic priority model. The operation of this model starts when such a group ($\Sigma\langle Jxt, Bg1 \rangle$) of jobs on such a machine come in P1, as in Figure 7. Now, transition (t1) is enabled by the readiness of the desired machine over f1 and the first priority rule information over f3. The result of firing t1 will cause $\Sigma\langle Jxt, Mg, Bg1, RLx1 \rangle$ to move to P2. The priority values RLx1 will work as filtering at t2. So the jobs with the highest priority values will pass to P3, while the rest jobs will go back on the path f7, P10, f31, f33 to place P1 again. However, if one job is passing to P3 then there is no problem and t9 will be enabled and fires to transfer this job through f35 to P9. The token in P9 will cause to enable t9_{ml} and so the job will pass to the machine level model.

If more than one job reach P3, then t3 will fire instead of t9. The transition (t3) works as the second filtering to select from the candidate jobs with respect to the second priority rule RLx2. The second priority rule is decided by f8 for all the related jobs in P3. The jobs that come out of firing t3 will be divided into two parts; one over f11 and the other over f12. It is clear that the transitions (t2, t3, t4, ... etc) work as filters to select a job with highest priority according to the sequence of the proposed priority rules. However, if there is no final decision to select only one job, then the selection will be done arbitrarily by pushing one job over f29 to P8 while the rest jobs will return to P10 and through it to P1 again. In this model the ADJ functions are used to reduce the value of the first priority rule (user priority) for the job already selected to be manufactured on the related machine. The importance of this reducing is to weak the weight of this priority, so if there are other similar jobs in P1 then these jobs will not be selected again according to their user's priorities. This will give a way to other jobs in the group to take their chances to be manufactured on the desired machine. Also it is noted from the model that the original user's priorities are fed from the knowledge base by firing t15 with each new group of jobs coming from P12. This will cancel the effecting of reduction done on this type of priority for the previous group of jobs. It is clear that the user's priority is fed to the model through the f55 from the place that represents the static knowledge base, since this priority is of predefined type. The other types of priorities are of heuristic type so they are fed from another place that represents the temporary knowledge base. This can be achieved by firing t16 through f58.

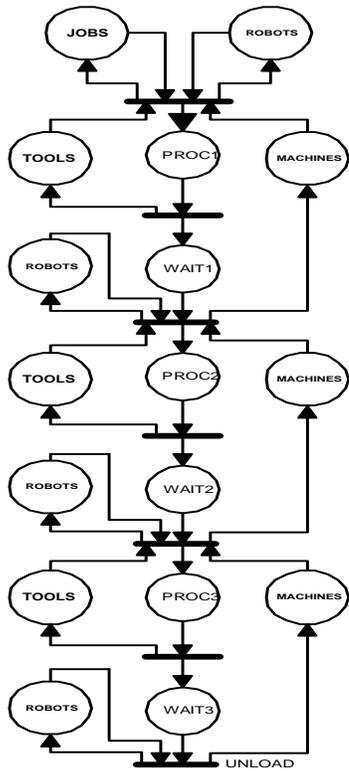


Figure 8. The CPN model of an automated manufacturing system.

6. System Evaluation

The following points are considered in the evaluation of the designed scheduler, these points are;

- the model capabilities to handle the complexity and the generality of the FMS systems,
- the flexibility and efficiency of the proposed heuristic priority algorithm, and
- the efficiency of the model’s monitoring facility.

The scheduler model presented in this paper has been compared with the model given in [17] to evaluate its capability. Figure 8 shows the FMS and its equivalent CPN model. This system has three machines (m1, m2, and m3), three robots (r1, r2, and r3), and a load/unload area. It is clear that such a model has lost the generality, because any change in the FMS resources or the manufacturing jobs route will change the model partially or completely. Moreover, the model suggested in [17] works at the machine-level only. In other word, this model will describe just one manufacturing cell. Hence, the advantages of the proposed model over the mentioned one is the flexibility to face the changes and the ability to represent multi cells manufacturing system based on a hierarchical scheme of cell and machine levels.

Two algorithms are used to control the sharing of the resources. These algorithms are the FIFO CPN model, and the heuristic priority policy CPN model. The generality of the FIFO algorithm is clear and simple so it does not need more explanations. The second algorithm is more important since it is used to schedule the jobs when they compete at the machines themselves. Our proposal suggests a sequence of six-priority levels, which can be changed in type and sequence partially or completely according to the expertise’s requirements. However, It is difficult to guess which combination of priority rules would lead to high performance. The following simulation will evaluate how the selecting sequence of these rules may affect the utilization of the manufacturing system.

Three measures of performance are tested for the manufacturing system evaluation [10], these measures are:

a). *Maximum flow time (Fmax):*

$$F_{max} = \max_j \{F_j\} \quad j = 1, 2, \dots, m$$

b). *Average flow time (Fav):*

$$F_{av} = \frac{1}{m} \sum_j F_j$$

where; F_j is the flow time for machine j , and m is the number of the machines.

c). *Machine utilization (Um):*

$$U_m = \frac{1}{m} \sum_j U_j, \text{ and } U_j = \frac{1}{F_j} \sum_{i \in m(j)} t_i$$

where; t_i is the processing time of operation (i).

$m(j)$: is the set of operations processed on machine (j).

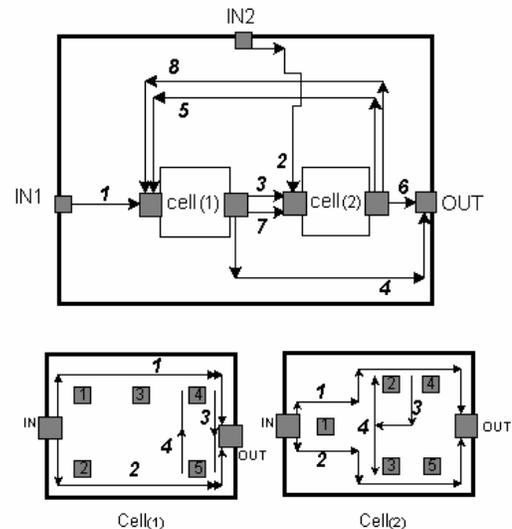


Figure 9. Case study layout.

The simulation results for the case study described in Figure 9, are given in Table 1. It is clear that there is no sequence of priority rules is dominant in terms of the solution quality for all problems. Hence, our proposal

will offer the expertise the ability to decide the desired sequence of these priorities according to the requirements. Since the FMS is of real time nature, the resources damaging and the response time to find an alternative is of critical significance. The designed model suggests software sensors that share with other input functions to enable the transitions. In fact, each software sensor interacts with a hardware sensor, which in turn is related to the working condition of such a resource in the system. However, these functions will not identify the type of the problem, but only indicate its appearance. Table 2 shows the response time for some simulated problems under the environment of the case study given in Figure 9.

Table 1. Simulation results of the priority sequences.

Test	Priority Sequence	Fmax (min)	Fav (min)	Um %
Cell (1)	PR1-PR2-PR3-PR4-PR5-PR6	458.4	454.0	97.5
	PR1-PR3-PR2-PR4-PR5-PR6	461.0	456.5	95.0
	PR1-PR4-PR3-PR2-PR5-PR6	463.5	456.2	96.3
	PR1-PR5-PR4-PR3-PR2-PR6	470.2	459.5	96.1
	PR1-PR5-PR3-PR2-PR4-PR6	462.3	454.1	93.2
	PR1-PR5-PR2-PR4-PR3-PR6	466.0	454.0	94.6
Cell(2)	PR1-PR2-PR3-PR4-PR5-PR6	430.5	423.7	98.0
	PR1-PR3-PR2-PR4-PR5-PR6	437.8	433.0	94.2
	PR1-PR4-PR3-PR2-PR5-PR6	443.8	429.2	95.9
	PR1-PR5-PR4-PR3-PR2-PR6	448.0	441.7	95.1
	PR1-PR5-PR3-PR2-PR4-PR6	451.2	439.6	95.3
	PR1-PR5-PR2-PR4-PR3-PR6	439.6	428.0	95.5

Time interval used in the test is 480 minutes.

Table 2. Recovery time for the damaged elements.

Test	Damaged Element	Response Time (sec)
Cell-level	LINK (3)	0.25
	LINK (4)	0.25
Machine-level	Machine(2) in Cell(1)	0.22
	Machine(2) in Cell(2)	0.21

7. Conclusions

On-line scheduling is an important task for obtaining efficiency and high productivity in flexible manufacturing systems. A hierarchical on-line scheduler model based on CPN has been presented in this paper. This model consists of two levels; the cell-level and the machine-level. It has been used to:

- interpret the output of the manufacturing route planner into graphical representation,
- achieve the sequencing, dispatching and monitoring the real-time implementation of the manufacturing plans, and
- generate feedback status to monitor any abnormal condition.

This scheduler has the following features;

1. The CPN models deal with multi-cell/multi-job manufacturing systems.
2. It has hierarchical design.
3. It monitors and controls the concurrency and synchronization between jobs to use limited resources.

The designed scheduler has been evaluated for different case studies to demonstrate its capabilities. The achieved results are very encouraging.



Kasim M. Al-Aubidy received his BSc and MSc degrees in control and computer engineering from the University of Technology, Iraq in 1979 and 1982, respectively, and the PhD degree in real-time computing from the University of Liverpool,

England in 1989. He is currently an associative professor in the Department of Computer Engineering at Philadelphia University, Jordan. His research interests include fuzzy logic, neural networks, genetic algorithm and their real-time applications. He is the winner of Philadelphia Award for the best researcher in 2000. He is also a member of the editorial board of the Asian Journal of Information Technology. He has coauthored 3 books and published 51 papers on topics related to computer applications.



Ali A. Ali received his BSc, MSc and PhD degrees in Electrical and computer engineering from the University of Technology, Iraq in 1980, 1987, and 1993, respectively. He is currently an assistant professor in the Department of Electrical and Computer Engineering at

Applied Science University, Jordan. His research interests include robotics and flexible manufacturing systems, modeling and simulation, and artificial intelligence. He has 6 published papers related to artificial intelligence, robotics and flexible manufacturing systems.

References

- [1] Al-Titnchi A. A. and Al-Aubidy K. M., "A Hierarchical Manufacturing Route Planner Design Based on Heuristic Algorithms", *Systems Analysis Modeling Simulation, SAMS Magazine*, vol. 42, pp.1119-1141, USA, 2002.
- [2] Barros T.C., de Figueiredo J.C., and Perkusich A., "A Fault Tolerant CPN Model for Flexible

- Manufacturing Systems", *Journal of the Brazilian Computer Society*, vol. 4, no. 2, November 1997.
- [3] Chen F.F., Chen J., "Performance Modelling and evaluation of dynamic tool allocation in FMSs Using CPNs: an Object-Oriented Approach", *Intr. Journal of Advanced Manufacturing technology*, vol. 21, no. 2, pp.98-109, January 2003.
- [4] Colombo A., and Carelli R., "Petri Nets for Designing Manufacturing Systems", Ch.11 in *Management and Control of Manufacturing Systems*, S. Tzafestas, Ed. Springer-Verlag, USA, 1996.
- [5] Feldmann K., Colombo A. W., Schnur C., and Stockel T., "Specifications, Design, and Implementation of Logic Controllers Based on CPN Models and The Standard IEC 1131, Part I: Specification and Design", *IEEE Trans. On Control Systems Technology*, vol. 7, no. 6, pp.657-665, November 1999.
- [6] Hatono I. and Tamura H., "Knowledge-Based On-Line Scheduling for Flexible Manufacturing", Ch.10 in *Soft Computing: Fuzzy Logic, Neural Networks & Distributed Artificial Intelligence*, Editors: F. Aminzadeh & M. Jamshidi, Prentice Hall, USA, 1997.
- [7] Jensen K., "Colored Petri Nets: Basic Concepts, Analysis, Methods & Practical Use", vol. 1 & 2, Springer-Verlag, USA, 1996.
- [8] Jorgensen J. B. and Kristensen L. M., "Computer Aided Verifications of Lamport's Fast Mutual Exclusion Algorithm Using Colored Petri Nets and Occurance Graphs with Symmetries", *IEEE Trans. On Parallel & Distributed Systems*, vol. 10, no.7, pp. 714-732, July 1999.
- [9] Kusiak A., "Modeling and Design of Flexible Manufacturing System", Elsevier Publisher 1986.
- [10] Kusiak A., "Manufacturing Systems: A Knowledge and Optimization-Based Approach", *Journal on Intelligent & Robotic Systems*, pp.27-50 March 1990.
- [11] Murata T., "Petri Nets: Properties, Analysis and Applications", *Proceedings of the IEEE*, vol. 77, no. 4, April 1989.
- [12] Peterson J. L., "Petri Nets Theory and The Modeling of Systems", Prentice Hall, USA, 1981.
- [13] Pettit IV R. G. and Gomaa H., "Integrating Petri Nets with Design Methods for Concurrent and Real-Time Systems", *Proceedings of the 2nd IEEE intr. Conf. on Engineering of Complex Computer Systems*, (ICECCS'96), USA 1996.
- [14] Qiao G., McLean C., and Riddick F., "Simulation System Modeling for Mass Customization Manufacturing", *Proceedings of the 2002 Winter Simulation Conference*, pp.2031-2036, USA, 2002.
- [15] Saitou K., Malpathak S., and Qvam H., "Robust Design of FMSs using CPN and Genetic Algorithm", *Journal of Intelligent Manufacturing*, no. 13, pp.339-351, 2002.
- [16] Song A., "Modeling and Control of On-Line Scheduling Systems in Flexible Manufacturing Environment", *2nd Intr. IEEE Conf. on CIM* pp.372-377, NewYork USA 1990.
- [17] Viswanadham N., "Performance Analysis of Automated Manufacturing Systems with Blocking and Deadlock", *2nd Intr. IEEE Conf. on CIM* pp.64-68, NewYork USA 1990.
- [18] Viswanadhan N. and Narahari Y., "Performance Modeling of Automated Manufacturing Systems", Prentice-Hall, USA 1994.

Appendix (A)

A.1 The cell-level model:

(a). The places definitions:

C(P1) = { < Jxs > } : This place represents jobs in the dummy input buffers (DI). The job (J) is denoted by x and s which mean the job's manufacturing stage number respectively.

C(R) = { < Ry > } : This place represents the availability of the robots in the FMS.

C(P2) = { < Jxs, INz, N > } : This place represents jobs in external input buffers (IN) of the FMS. The N value is the job's priority according to the FIFO policy.

C(CAP) = { < Le0 > } : This place represents the availability of the transportation links such as conveyor, AGV, ...etc.

C(LNK) = { < Jxs, Le0, N > } : This place represents the jobs at initial position on the related link.

C(E) = { < Jxs, Lew, N > } : This place represents the movement values of the jobs on the related link. This movement is scaled by the parameter (w).

C(LOG) = { < Le0 > } : This place represents the logical availability of the links at their initial positions.

C(P4) = { < Jxs, Ba1, N > } : This place represents jobs in the cells input buffers. Where a means the related cell and (1) means input buffer.

C(P5) = { < Jxs, Ca, N > } : This place represents jobs inside the related cell (Ca).

C(P6) = { < Jxs, Ba2, N > } : This place represents jobs in cells output buffers. Where a means the related cell and (2) means output buffer.

C(P7) = { < Jxs, Ob, N > } U { < Jxs, Df, N > } : This place represents jobs in external output buffer (Ob) or an intermediate storage buffer (Df) of the FMS.

C(P8) : This place is equivalent to C(P1) but now it represents dummy output buffers (OD).

These places represent the availability of the buffers at the cell level.

C(AV1) = { < INz > }

C(AV2) = { < Ba1 > }

C(AV3) = { < Ba2 > }

C(AV4) = { < Ob > } U { < Df > }

C(K-B) = { < INz > , < Le0 > , < Ba1 > , < Ba2 > , < Df > , < Ob > } : This place represents the source of information about the types and the capacities of the resources

(b). The transitions definitions:

$$C(t1) = \{ \langle Jxs, Ry \rangle \}$$

$$C(t2) = \{ \langle Jxs, INz, Ry, Le0 \rangle \}$$

The following gates control the transportation mechanism, where # represents the actual length of the related transportation link;

$$C(t3) = C(t4) = C(t6) = C(t7) = \{ \langle Le0 \rangle \}$$

$$C(t5) = \{ \langle Lew \rangle \}$$

$$C(t8) = \{ \langle Jxs, Le#, Ry \rangle \}$$

$$C(t9) = \{ \langle Jxs, Ba1, Ry \rangle \}$$

$$C(t10) = \{ \langle Jxs, Ca, Ry \rangle \}$$

$$C(t11) = \{ \langle Jxs, Ba2, Ry, Le0 \rangle \}$$

$$C(t12) = \{ \langle Jxs, Le#, Ry \rangle \}$$

$$C(t13) = \{ \langle Jxs, Df, Ry, Le0 \rangle \}$$

$$C(t14) = \{ \langle Jxs, Ob, Ry \rangle \}$$

$$C(t15) = \{ \langle INz, \langle Le0 \rangle, \langle Ba1 \rangle, \langle Ba2 \rangle, \langle Df \rangle, \langle Ob \rangle, \langle Ry \rangle \}$$

A.2 The machine-level model:*(a). The places definitions:*

$C(P1) = \{ \langle Jxt, N \rangle \}$: This place represents jobs that are ready to be processed inside the related cell. The job (J) is denoted by x and t which are the job number and manufacturing operation number.

These places represent the availability of buffers, machines and machines tools inside the related manufacturing cell

$$C(AV1) = \{ \langle Bg1 \rangle \}$$

$$C(AV2) = \{ \langle Mg \rangle \}$$

$$C(AV3) = \{ \langle Bg2 \rangle \}$$

$$C(AV4) = \{ \langle Tk \rangle \}$$

$$C(TOL) = \{ \langle Ygy \rangle \}$$

$C(K-B) = \{ \langle Lr0 \rangle, \langle Bg1 \rangle, \langle Bg2 \rangle, \langle Mg \rangle, \langle Tk \rangle, \langle Ygy \rangle \}$: This place represents the source of information about the types and the capacities of the resources inside the related cell.

(b). The transitions definitions:

$$C(t1) = \{ \langle Jxt, Lr0, Rq \rangle \}$$

These gates control the transportation mechanism, where # represents the actual length of the related transportation link.

$$C(t2) = C(t3) = C(t5) = C(t6) = \{ \langle Le0 \rangle \}, C(t4) = \{ \langle Lrw \rangle \}$$

$$C(t7) = \{ \langle Jxt, Lr#, Bg1, Rq \rangle \}$$

$$C(t8) = \{ \langle Mg, Bg1 \rangle \}$$

$$C(t9) = \{ \langle Jxt, Mg, Ygy, Bg1, Rq \rangle \}$$

$$C(t10) = \{ \langle Jxt, Mg, Ygy, Rq \rangle \}$$

$$C(t11) = \{ \langle Jxt, Bg2, Lr0, Rq \rangle \}$$

$$C(t12) = \{ \langle Jxt, Lr#, Rq \rangle \}$$

$$C(t13) = \{ \langle Jxt, Tk, Lr0, Rq \rangle \}$$

$$C(t14) = \{ \langle Mg, \langle Lr0 \rangle, \langle Bg1 \rangle, \langle Bg2 \rangle, \langle Ygy \rangle, \langle Tk \rangle, \langle Rq \rangle \}$$

A.3 The FIFO model:*(a) The places definitions*

$C(FIFO1) = \{ \langle Jxs, Ry \rangle \}$ at cell-level
 $C(FIFO1) = \{ \langle Jxt, Rq \rangle \}$ at machine-level
 $C(INC) = \{ \langle ?, N \rangle \}$: This place represents the instant priority value (N) related to the argument (?), where ? is INz, Ba1, Ba2, Ob, or Df at cell-level, and it is Bg2 or Tk at machine-level.

(b) The transitions definitions

$$C(tag) = \{ \langle Jxs, Ry, ? \rangle \}$$
 at cell-level

$$C(tag) = \{ \langle Jxt, Rq, ? \rangle \}$$
 at machine-level

$C(CAP) = \{ \langle Lr0 \rangle \}$: This place represents the availability of related transportation link such as conveyor, AGV, ...etc.

$C(LNK) = \{ \langle Jxt, Lr0, N \rangle \}$: This place represents jobs at the initial position on the related link.

$C(E) = \{ \langle Jxt, Lrw, N \rangle \}$: This place represents the movement values of the jobs on the related link, which is scaled by the parameter (w).

$C(LOG) = \{ \langle Lr0 \rangle \}$: This place represents the logical availability of the links at their initial positions.

$C(P2) = \{ \langle Jxt, Bg1 \rangle \}$: This place represents the jobs in the machines i/p buffers, where g means the related machine and (1) the i/p buffer.

$C(M) = \{ \langle Jxt, Mg, Ygy \rangle \}$: This place represents jobs under processing on the related machine (g) and the tool (y).

$C(P3) = \{ \langle Jxt, Bg2, N \rangle \}$: This place represents jobs in the machines o/p buffers, where g means the related machine and (2) the o/p buffer.

$C(P4) = \{ \langle Jxt, Tk, N \rangle \}$: This place represents jobs in an intermediate storage buffer (Tk) within the manufacturing cell.

$C(P5) = \{ \langle Jxt \rangle \}$: This place represents jobs that are ready to be transferred to another manufacturing cell.

$C(R) = \{ \langle Rq \rangle \}$: This place represents the robots inside the related manufacturing cell.

A.4 The heuristic priority algorithm model:*(a) The places definitions*

$C(P1) = \{ \langle Jxt, Bg1 \rangle \}$: This place represent groups of jobs in dummy input buffers of the manufacturing machines. This dummy buffer denotes logical location contains the jobs that the heuristic priority algorithm selects from them to use the related machine.

$C(P2) = \{ \langle Jxt, Mg, Bg1, RLx1 \rangle \}$: This place represents the jobs ready to be manufactured on the related machine according to the first priority rule (RLx1)

$C(P3) = C(P4) = C(P5) = C(P6) = C(P7) = C(P8) = C(P10)$: These places have similar color tokens as $C(P2)$. They contains jobs coming through the filters (the transitions) according to the next priority rules [2,3,4,5].

$C(P9) = \{ \langle Jxt, Mg, Bg1 \rangle \}$: This place represents the selected job to be manufactured on the related machine.

$C(P11) = C(P12) = \{ \langle Jxt, Bg1 \rangle \}$: These places represent the new groups of jobs in the input buffers of the related machines. They are used to initiate the user priority.

These places represent the priority values according to the rules of the heuristic priority algorithm.

$$C(PR1) = \{ \langle RLx1, Mg \rangle \}$$

$$C(PR2) = \{ \langle RLxt2, Mg \rangle \}$$

$$C(PR3) = \{ \langle RLxt3, Mg \rangle \}$$

$$C(PR4) = \{ \langle RLxt4, Mg \rangle \}$$

$$C(PR5) = \{ \langle RLxt5, Mg \rangle \}$$

$C(GAR) = \{ \langle RLx1, Mg \rangle \}$: This place works as garbage for the user priority tokens of the jobs, which are already manufactured on the related machine.

These places are the sources of information about the values of priorities for the jobs according to the heuristic priority rules.

$$C(SK-B) = \{ \langle RLx1, Mg \rangle \}$$

$$C(TK-B) = \{ \langle RLxt2, Mg \rangle \} \cup \{ \langle RLxt3, Mg \rangle \} \cup \{ \langle RLxt4, Mg \rangle \} \cup \{ \langle RLxt5, Mg \rangle \}$$

(b) The transitions definitions

All the transitions in this model has the same definition which is $\{ \langle Mg \rangle \}$.