International Journal of Industrial Engineering & Production Research

June 2011, Volume 22, Number 2
pp. 115-121

http://IJIEPR.iust.ac.ir/

ISSN: 2008-4889

Time Management Approach on a Discrete Event Manufacturing System Modeled by Petri Net

Meysam Zareiee, Abbas Dideban* & Ali A. Orouji

Meysam Zareiee *is a PhD student at the Faculty of Electrical and Computer Engineering, Semnan University, Semnan, IRAN*
****Abbas Dideban*** *is an assistant professor at the Faculty of Electrical and Computer Engineering, Semnan University, Semnan, IRAN*
Ali A. Ourouji *is an associate professor at the Faculty of Electrical and Computer Engineering, Semnan University, Semnan, IRAN*

KEYWORDS

Discrete event system,
Supervisory control,
Petri Net, Constraint

ABSTRACT

This paper presents a method to manage the time in a manufacturing system for obtaining an optimized model. The system in this paper is modeled by the timed Petri net and the optimization is performed based on the structural properties of Petri nets. In a system there are some states which are called forbidden states and the system must be avoided from entering them. In Petri nets, this avoidance can be performed by using control places. But in a timed Petri net, using control places may lead to decreasing the speed of systems. This problem will be shown on a manufacturing system. So, a method will be proposed for increasing the speed of the system without using control places.

1. Introduction

Discrete event systems are the systems which their states can be changed by the occurrence of events. To control these systems, supervisory control theory is proposed by Ramadge and Wonham [1], [2]. The base of this theory is to restrict the behavior of system for obtaining desirable function. This can be achieved by disabling some controllable events in special conditions [3]. For applying this theory, automata was the first tool for the modeling of discrete event systems. But when the number of states and events in the system is large, modeling systems based on this tool is very difficult or may be impossible [4]. So, Petri net (PN) was proposed as an alternative tool for modeling discrete event systems to overcome this problem [5]. Compact structure, mathematical and structural properties have made PN a suitable tool for modeling discrete event systems. This tool can be divided into two groups of timed PN and untimed PN and is composed of places, transitions and arcs. When a system is modeled by PN, the events are assigned to its transitions. But in a system for some events it is possible to be uncontrollable. So the supervisor cannot manipulate any functions on them and these events may lead to undesirable function. So, in [6] a method is proposed which decrease the space of authorized states to obtain the desirable function.

In a system it is possible for some state to be called forbidden states which the system must be avoided from entering them. For preventing the system from entering these states, some methods are proposed. In [7] a method is proposed for preventing the system from entering the forbidden states by assigning some conditions to the controllable transitions. These conditions are calculated online and lock the controllable transitions in some special states. A similar method for assigning conditions to the transitions is proposed in [8], [9], but in this method the conditions are calculated offline.

Another method for preventing the system from entering the forbidden states is adding some places to the PN model. These places are called control places. In [10], a method is presented for calculating control places. In this method control places are calculated from forbidden states.

For obtaining desirable function, applying some inequalities on the system is possible. These

* **Corresponding author. Abbas Dideban**
Email: *adideban@semnan.ac.ir*
Paper first received Dec. 11. 2010, and in revised form May. 15. 2011.

inequalities work as constraints and can be obtained from the forbidden states [11]. These inequalities can be enforced on the system by the method presented in [12]. But applying this method on the timed PNs, doesn't give the optimal operation since it is possible for the method to spend extra time for preventing the system from entering the forbidden states.

In this paper the objective is to show that the idea in [12] doesn't give the least cycle time for the operation of system. This concept will be shown by a simple example. So, in this example we will propose a method for obtaining the desirable function by management of time. In this method some conditions will be defined and the related controllers will be applied on the controllable transitions of system to allow firing of the transitions in special states. In this example it will be obvious that the cycle time is smaller than the last method.

The rest of this paper is as follows. In section 2, some important definitions and concepts will be presented. The previous methods for obtaining the desirable function of system will be recalled in section 3. The main idea in this paper for management of the time will be presented in section 4. Finally conclusion is discussed in section 5.

2. Preliminary Presentation

In this section the goal is to introduce some concepts that are necessary for presenting the new approach in this paper. It is supposed that the reader is familiar with the PN basis [13].

PNs are composed of places, transitions and arcs. Places are shown with circles and the transitions with bars. Places and transitions are connected together by arcs.

A place can be empty or marked. When a place is marked, there is a token or there are some tokens in it. A transition can fire when all the input places are marked. When a transition fire, a token is eliminated from all the input places and a token is added to all the output places [13].

The relation between places and transitions are shown by incidence matrix. This matrix is related to the marking of system. This matrix is a $n \times m$ matrix where n is the number of places and m is the number of transitions. In this matrix, when a place P_i is an input of a transition T_j, the junction of ith row and jth column is -1 and if this place is the output of that transition this junction is 1 and if there is not an arc between them, this junction is zero. For example consider the PN model in Fig. 1. This model has three places (P_1 and P_2 and P_3) and three transitions (t_1 and t_2 and t_3). The incidence matrix for this model is as follows:

$$W = \begin{bmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix}$$

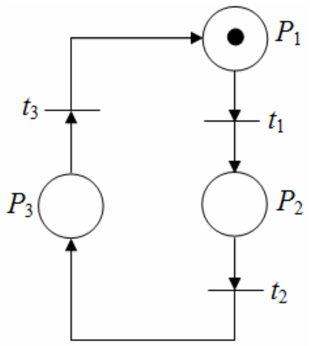

Fig. 1. A PN model.

When the number of tokens is not greater than one, PN is called safe, otherwise it is called non safe. In this paper we focus on safe PN. The state of each PN can be shown by writing the places that are marked. For example suppose that in a state, the places P_1 and P_2 and P_3 are marked and the other places are empty. So the state of this PN is $P_1P_2P_3$. The state of PN can be changed by firing of transitions. These changes are shown by marking graph of PN. Marking graph is a graph that illustrates all the states and shows that firing of which transitions create each state. The reachability graph for the PN model in Fig. 1 is depicted in Fig. 2.

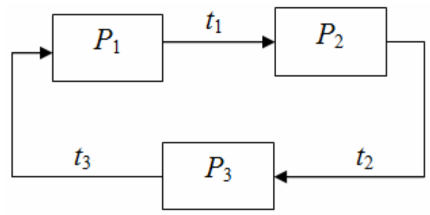

Fig. 2. The marking graph of the PN in Fig. 1.

After modeling systems by PNs, for safety operation of system, it is necessary to apply some conditions on the system. These conditions are called specification. In the model of system, there are some states that don't respect to the specifications or are deadlock states (when system is in a deadlock state, it is locked). These states are forbidden states and must be prevented. In a safe PN, it is possible to assign a constraint in the form of an inequality to each forbidden state [11]. These inequalities can be enforced on the system by adding a control place instead of each inequality [12]. The method for calculating the control places is described in section 2.1.

2.1. Calculating the Control Places

To calculate the control places, suppose that the set of constraints is shown as follows:

$$L.\, M_P \leq b$$

Where M_P is the marking vector ($M_P^T=[m_1\ m_2\ \dots\ m_n]$ where m_i is the number of token in place P_i), L is a $n_c \times n$ matrix, b is a $n_c \times 1$ vector, n_c is the number of constraints and n is the number of places. In this method instead of each constraint, a control place is added to the system. These control places are calculated by converting inequalities to equalities. So, when the control places are added to the system, its incidence matrix changes and instead of each constraint (control place), a row is added to this incidence matrix. Now, suppose that the incidence matrix of the system before enforcing the control places is shown by W_P. Also suppose that the rows that must be added to this matrix are shown by W_c. So, this matrix is calculated as follows:

$$W_c = -L.W_P$$

This matrix must be added to the system, and the incidence matrix for controlled model is obtained as follows:

$$W = \begin{bmatrix} W_p \\ W_c \end{bmatrix}$$

After this calculation, the initial marking of these control places must be calculated. For this reason suppose that the initial marking of system is shown by M_{P0}, then the initial markings of the control places are calculated as follows:

$$M_{s0} = b - L.M_{P0}$$

So the initial marking of the controlled model is as follows:

$$M_0 = \begin{bmatrix} M_{p0} \\ M_{s0} \end{bmatrix}$$

3. System Control Using Known Methods

In [14], a method has been introduced that using it, in some conditions, it is possible to control systems with a compact controller.

This method uses the controller presented in [12] for controlling systems and convert this controller to a compact controller. But using this controller leads to eliminating some states that cannot be forbidden states in some moments and then leads to decreasing the speed of system. However there is not a method to solve this problem. But we will show that in timed PN, it is possible to solve this problem to increase the speed of system. In this section, the goal is to develop the idea in [12] and [14] for reaching to an efficient model in timed PN. Before introducing the new idea, we recall the example presented in [14] for the timed PN model of system to introduce the problem of controller by the last method.

Example 1 [14]: A manufacturing system is composed of two independent machines M_1 and M_2, one transfer robot of the parts and one test bench where the final products are tested. Each machine has the following operating cycle: By occurrence of the event t_i, the machine starts working. When the work is finished (occurrence of event t_2 and t_5), the produced part is transferred by the robot on the test bench, and by occurrence of event t_3 and t_6 a new cycle can be started again. There are two types of events in this system: the controllable events and uncontrollable events. Only events t_1 and t_4 are controllable:

$$T_c = \{t_1, t_4\} \text{ and } T_u = \{t_2, t_3, t_5, t_6\}.$$

The process model of this system is shown in Fig. 3. According to this figure, a token in place P_i will be enabled for the output transitions of this place if it has stayed for d_i time units.

In this example, the product of machine M_1 must be coupled with the product of machine M_2. So, firstly the robot must transfer the product of machine M_1 and then the product of machine M_2. Therefore, the specification model of this system is shown in Fig.4.

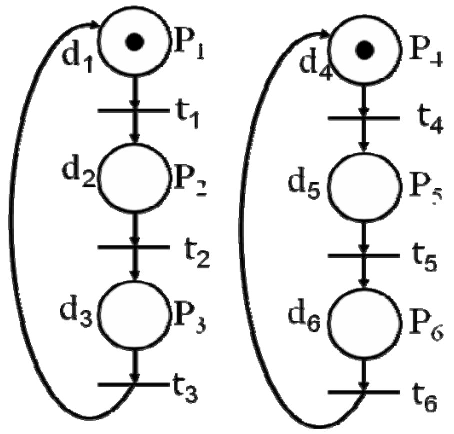

Fig. 3. The PN model of the system in example 1.

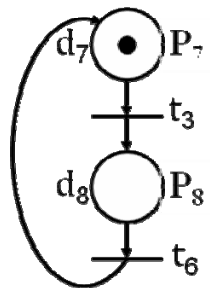

Fig. 4. The specification model of the system in example 1.

This model of specification must be synchronized with the model of system. The closed loop model of this system is illustrated in Fig. 5. This model is called Quasi PN. A Quasi-PN is a PN which respects the following rules of firing:

1. A controllable transition is firable in the same way as in a ordinary PN.
2. An uncontrollable transition is firable if all its input places belonging to the process are marked.

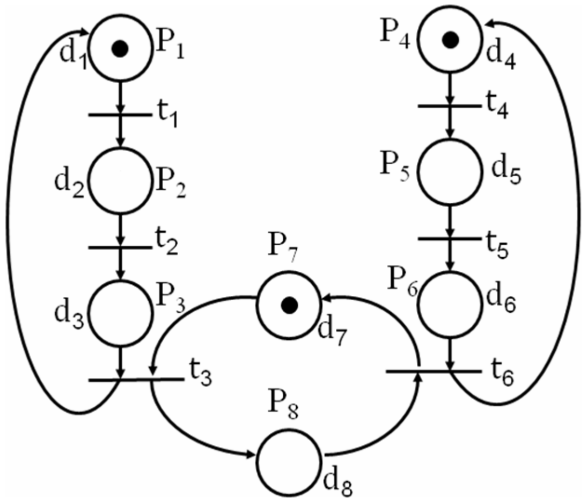

Fig. 5. The closed loop model of the system in example 1

So, it is possible to find forbidden states from the difference between Quasi PN and ordinary PN. If in a state an uncontrollable transition is firable for Quasi PN model but not for ordinary PN model, this state is forbidden. To see this concept, let us to show the marking graph of this system. This marking graph is illustrated in Fig. 6.

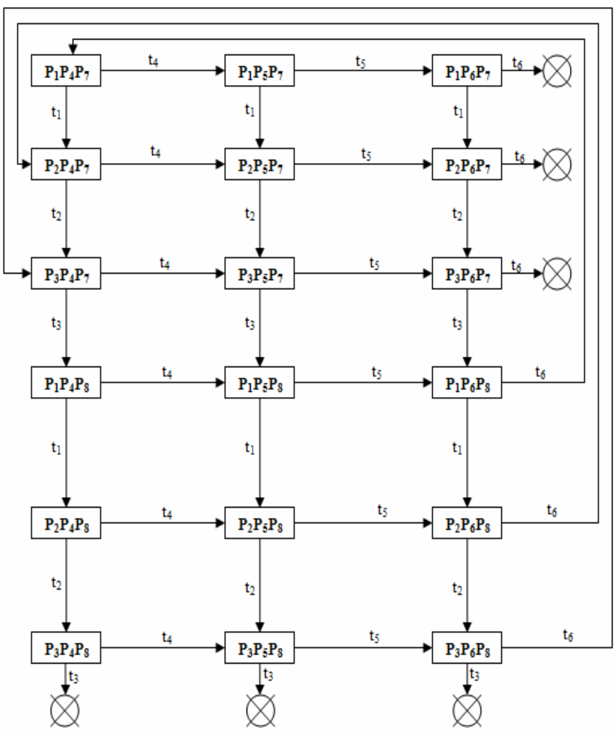

Fig. 6. The marking graph of the model of system in example 1.

As it is obvious from marking graph, the states $P_1P_6P_7$, $P_2P_6P_7$, $P_3P_6P_7$, $P_3P_4P_8$, $P_3P_5P_8$, $P_3P_6P_8$ are forbidden states [14]. So, for controlling the system, we must prevent it from entering these states. But all of these states are obtained by firing of uncontrollable events. Then we cannot prevent firing of these transitions. But it is possible to find some states that preventing them leads to preventing the forbidden states. These states are called border forbidden states. Border forbidden states are the states that are obtained by firing of controllable events. For this example, these states are $P_2P_6P_8$, $P_2P_5P_8$, $P_2P_4P_8$, $P_3P_5P_7$, $P_2P_5P_7$, $P_1P_5P_7$ [14]. So, by preventing the system from entering the border forbidden states, it doesn't reach to any of the forbidden states.But in safe PN, it is possible to assign some constraints to border forbidden states [11]. These constraints are as inequalities. When these inequalities are satisfied by a controller, the forbidden states cannot be occurred. For constructing these constraints suppose that the state $P_1P_2\ldots P_n$ is a border forbidden state. So the constraint related to this state is:

$$m_1+m_2+\ldots+m_n \le n-1$$

For this example the constraints related to the border forbidden states are as follow:

$m_1+m_5+m_7\le 2$, $m_2+m_5+m_7\le 2$,

$m_3+m_5+m_7\le 2$, $m_2+m_4+m_8\le 2$,

$m_2+m_5+m_8\le 2$, $m_2+m_6+m_8\le 2$.

For enforcing the constraints on the system, it is possible to add a control place instead of each constraint [12]. Therefore, for enforcing these constraints on the system, 6 places must be added to PN model of the system. But these constraints can be converted into two constraints that enforcing the two new constraints on the system, prevents it from entering all the forbidden states [14]. So these constraints can be reduced into two constraints as follow:

$$m_2+m_3+m_8\le 1 \quad , \quad m_5+m_6+m_7\le 1$$

Then, the control places related to these constraints can be enforced on the system. The controlled model of this example after adding the control places are shown in Fig. 7.

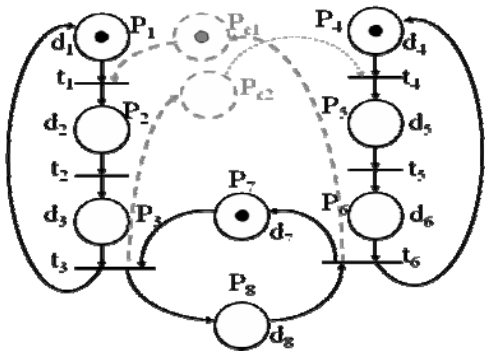

Fig. 7. The controlled model of the system in example 1.

But this model has a big problem since by using this method, some states are forbidden which can be authorized in some moments (border forbidden states). This concept shows that by using this method, the speed of system decreases. For example, in this case the state $P_1P_6P_7$ is a forbidden state (without considering control places). To forbid this state, when the system is in the state $P_1P_4P_7$, the transition t_4 must not fire until t_3 fires. Fig. 7 verifies this concept. So, before firing t_3, the places P_5 and P_6 cannot be marked. It means that the machine M_2 cannot begin working (P_5) and the product of M_2 cannot be transferred by the robot on the test bench (P_6) until t_3 fires. This concept decreases the speed of system because the machine M_2 must wait until the machine M_1 complete its work and its product be transferred to the test bench then it can start his work.
But it is possible that completion the task of M_2 and transfering its product take longer time comparing with the first machine, naturally. In this case the machine M_1 and M_2 can start their tasks simultaneously. In the next section we introduce a new idea to solve this problem in timed PN.

4. An Efficient Controller in Timed PN

In previous section, we saw that how the control places decrease the speed of the system. This problem is occurred when the system has uncontrollable transitions. In this section we want to increase the speed of system. This concept is obtained in timed PN by authorizing the forbidden states in some moments. For reaching to this goal, the conditions and their related controllers can be added to the controllable transitions (Fig. 8).
The base of this idea is that a supervisor monitors the behavior of system and in some conditions it permits firing of the controllable transitions and in the other conditions doesn't permit.
To explain this concept, at first step suppose that the system is in the state $P_1P_4P_7$ and $d_1+d_2+d_3 < d_4+d_5+d_6$ (don't consider the control places).
In this case both of the transitions t_1 and t_4 can fire But in this state if $d_1+d_2+d_3 > d_4+d_5+d_6$, t_1 can fire and t_4 cannot fire (firing t_4 leads to violating the specification). However after passing some seconds, firing of t_4 is possible. For example suppose that $d_1^{'}$ be the residual time for the token in place P_1 to be enabled for the transition t_1. In this case when $d_1^{'}+d_2+d_3 < d_4+d_5+d_6$, t_4 can fire by verifying the specification. Now for example 1 we can define two conditions for t_1 and t_4 and by applying the controllers related to these conditions, the best performance will be obtained. By applying these controllers, the efficiency of the system increases and the manufacturing time decreases. In the follwing we introduce these conditions and the related controllers for obtaining an efficient model.

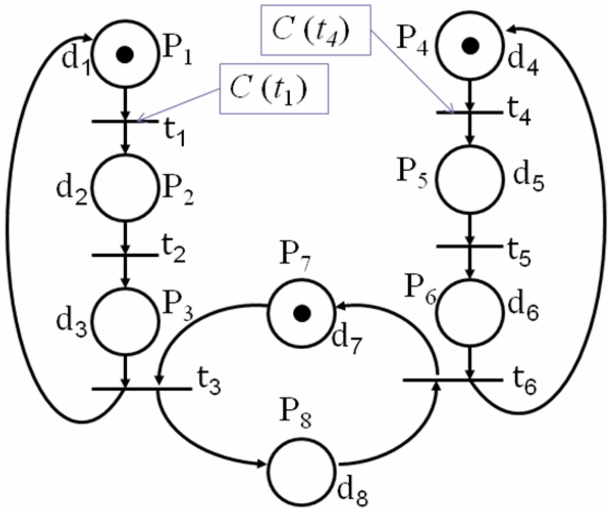

Fig. 8. Considering conditions on the model of system in example 1.

The condition for transition t_1 can be defined as follows :

Con_1 = (M_i== $P_1P_4P_8$) && ($d_1+d_2+d_3 > d_4^{'}+d_5+d_6$) || ($M_i$== $P_1P_5P_8$) && ($d_1+d_2+d_3 > d_5^{'}+d_6$) || ($M_i$== $P_1P_6P_8$) && ($d_1+d_2+d_3>d_6^{'}$) || ((M_i!= $P_1P_4P_8$) && (M_i!= $P_1P_5P_8$) && (M_i!= $P_1P_6P_8$)).

where $d_i^{'}$ is the residual time for a token in place P_i to be enabled for the output transitions of this place.
This condition considers the states and the momonets that the transition t_1 can fire without violating the specification. According to this condition, locking the transition t_1 is determined by a controller as follows:

$$C(t_1) = \begin{cases} Disable & \text{If } Con_1 = 0 \\ Enable & \text{If } Con_1 = 1 \end{cases}$$

Applying $C(t_1)$ on the transition t_1 leads to the best firing performance of this transition.
Similare to the transition t_1, the condition for transition t_4 is defined as follows:

Con_4= (M_i==$P_1P_4P_7$) && ($d_1^{'}+d_2+d_3 < d_4+d_5+d_6$) || ($M_i$==$P_2P_4P_7$) && ($d_2^{'}+d_3 < d_4+d_5+d_6$) || ($M_i$==$P_3P_4P_7$) && ($d_3^{'}< d_4+d_5+d_6$) || (($M_i$!=$P_1P_4P_7$) && ($M_i$!=$P_2P_4P_7$) && ($M_i$!=$P_3P_4P_7$)).

This condition considers the states and the moments that the transition t_4 can fire without violating the specification. According to this condition, locking the transition t_4 is determined by another controller as follows:

$$C(t_4) = \begin{cases} Disable & \text{If } Con_4 = 0 \\ Enable & \text{If } Con_4 = 1 \end{cases}$$

Applying $C(t_4)$ on the transition t_4 leads to the best firing performance of this transition.

By applying the controllers $C(t_1)$ and $C(t_4)$ on the transitions t_1 and t_4 respectively, the best operation of the system can be obtained (Fig. 8). Using this method, some states which were forbidden in long period of time, are forbidden in a smaller period of time than the past. This concept increases the speed of the system. Now, we want to apply these controllers on example 1. For this reason, suppose that d_1=0, d_2=2, d_3=3, d_4=0, d_5=2, d_6=2, d_7=0, d_8=0. Now we calculate the time for a cycle of manufacturing process for both methods (the last method and the new method) and then compare them. So, we make the marking graph of this system for a manufacturing cycle of each machine. Using the previous method [14], this marking graph is shown in Fig. 9. In this figure, the number on each arc, represents the necessary time for changing the state of system from a state to another state.

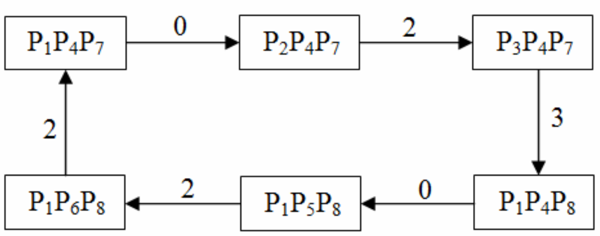

Fig. 9. The marking graph of the controlled system in example 1 by existing control places.

As it is obvious in Fig.9, the time for completing the task is 9 time units.

The marking graph after using the new method is illustrated in Fig. 10. In this figure ε has a very small value and can be neglected.

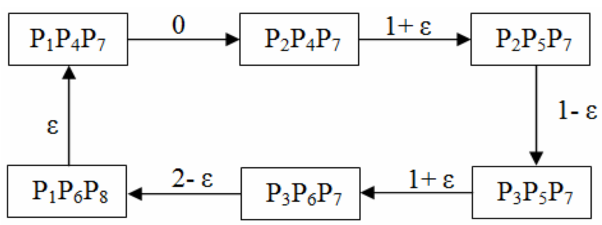

Fig. 10. The marking graph of the controlled system in example 1 by using condition 1 and condition 2.

In Fig. 10, it is obvious that the time for completing the task of each machine in a cycle is 5 (by neglectin ε). It is clear that this time is very smaller than the time using the previous method [14]. This example shows the capability of the new method for designing an efficient controller. In this example, the speed of the system has improved. This improvement is because of the authorizing of forbidden states in some moments. In fact, the forbidden states are forbidden in some moments and in other moments they are authorized. After modeling system by PN and finding forbidden states, we must not forbid these states for all of the times but only for some moments. It means that the forbidden states are provisionally forbidden states.

As it is obvious from this example, using control places in timed PN may lead to decreasing the speed of system since the control places don't permit working of a machine before completing the task of another machine. This concept decreases the speed of system. But by considering conditions on the controllable transitions and applying the related transition controllers, this problem can be solved. These transition controllers increase the speed of system because they permit working of a machine before completing the task of another machine without violating the specification.

This method is a very efficient method in manufacturing systems and can be generalized for all the systems which are modeled by PNs. So, in manufacturing systems, it is not necessary to forbid all the forbidden states for all the times. They must be provisionally forbidden states.

5. Conclusion

In this paper we have presented a method for increasing the speed of a manufacturing system modeled by timed PN. We have explained that adding control places to the timed PNs for controlling systems leads to decreasing the speed of systems. Since these places do not consider the time and prevent the firing of some transitions when it is not necessary, they decrease the speed of system. But for obtaining an efficient operation, we consider some transition controllers on the controllable transitions to prevent the firing of them in some moments and to permit this firing in other moments. This concept leads to increasing the speed of system and obtaining an acceptable operation. This concept can be generalized for all the manufacturing systems.

References

[1] Ramadge, P.J., Wonham, W.M., *"Modular Feedback Logic for Discrete Event Systems*", SIAM Journal of Control and Optimization, 25(5), 1987, pp. 1202–1218.

[2] Ramadge, P.J., Wonham, W., *"The Control of Discrete Event Systems"*, Dynamics of discrete event systems [Special issue]. Proceedings of the IEEE, 77(1), 1989, pp. 81–98.

[3] Cassandras, G.G., Lafortune, S., *Introduction to Discrete Event Systems*, Springer, 2008.

[4] Moody, J.O., Antsaklis, P., "*Petri Net Supervisor for DES with Uncontrollable and Unobservable Transition*", IEEE Trans. Automatic Control, 45(3): 462-476, mars, 2000.

[5] Krogh, B.H., Holloway, L.E., *"Synthesis of Feedback Control Logic for Discrete Manufacturing Systems",* Automatica, Vol. 27, No. 4, 1991, pp. 641–651.

[6] Kumar, R., Holloway, L.E., "*Supervisory Control of Deterministic Petri Nets with Regular Specification Languages"*, IEEE Trans. Automatic Control, 41(2): 1996, pp. 245-249.

[7] Holloway, L.E., Guan, X., Zhang, L., *"A Generalisation of Sstate Avoidance Policies for Controlled Petri Nets"*, IEEE Trans. Automatic Control, AC-41, 6, 1996, pp. 804- 816.

[8] Dideban, A., Alla, H., "*Solving the Problem of Forbidden States by Feedback Control Logical Synthesis*", The 32nd Annual Conference of the IEEE Industrial Electronics Society, Paris, FRANCE, 2006.

[9] Dideban, A., Alla, H., "*Feedback Control Logic Synthesis for Non Safe Petri Nets*", The 13th IFAC symposium on information control problems in manufacturing, Moscow, Russia. 2009.

[10] Ghaffari, A., Rezg, N., Xie, X.-L, *"Design of Live and Maximally Permissive Petri Net Controller using the Theory of Regions*", IEEE Transactions on Robotics and Automation, 19(1), 2003.

[11] Giua, A., Petri Nets as Discrete Event Models for Supervisory Control. Ph.D. *Thesis*, 1992.

[12] Yamalidou, K., Moody, J., Lemmon, M., Antsaklis, P, *"Feedback Control of Petri Nets Based on Place Invariants*", *Automatica*, 32(1), 1996, pp. 15–28.

[13] David, R., Alla, H.. "*Discrete, continuous, and hybrid Petri Nets"*, Springer, 2005.

[14] Dideban, A., Alla, H.. "*From Forbidden State to Linear Constraints for the Optimal Supervisory Control*", Control Engineering and applied Informaics (CEAI), 7(3), 2005, pp. 48-55.