SYNTHESIS METHOD OF A PN CONTROLLER USING FORBIDDEN TRANSITIONS SEQUENCES

R. Bekrar, N. Messai, N. Essounbouli, A. Hamzaoui and B. Riera
CReSTIC, IUT de Troyes, 9 rue de Qubec BP 396, 10026 Troyes Cedex, France

Keywords: Forbidden state problem, Forbidden transitions sequences, Supervisory control, Discrete event systems, Petri Nets.

Abstract: In this paper, we propose a control synthesis method for Discrete Event Systems (DES) modelled by a bounded ordinary Petri Nets (PN) to treat a forbidden state problem. The considered PN is transitions controllable and contains measurable and non measurable places. The PN controller is synthesised using the forbidden transitions sequences. The latter, are deduced from the PN reachability graph and considered as a forbidden language generated by the PN model. To show the efficiency of the proposed method an illustrative example is presented.

1 INTRODUCTION

In Discrete Event Systems (DES), it is important to prevent the system to reach undesirable states. Hence, given a DES model and a specification of the desired behaviours, one must to synthesise an efficient controller in order to achieve this goal. Among tools used to synthesis a DES controller, Petri Nets (PN) have been successfully considered as an efficient formalism for DES control.

This paper is considered as the second phase of our work (Bekrar et al., 2006a; Bekrar et al., 2006b) on the identification of a DES. It allows to complete the identified model by adding control places. Indeed, the identified PN model can generate all the DES states. However, it can generates, sometimes, some states which are not observed during the system functioning. Thus, our objective is to synthesise a PN controller to prevent the reachability of the forbidden states and guarantees the desired ones.

The synthesis problem of PN controllers has been widely treated in the literature. It was introduced either as a forbidden states problem (FSP) or a forbidden state transitions problem (FSTP). The synthesis methods of a PN controller allow to add a set of control places to the initial model in order to generate the desired behaviours. Several approaches are suggested in order to solve this problem. Among them we can find, the logical predicate based solutions (Krogh and Holloway, 1991; Boel et al., 1995; Holloway et al., 1996). The key idea is to perform an off-line structural analysis for determining algebraic expressions. The latter will be evaluated on-line for the current marking in order to decide the firing of the controllable transitions. These approaches are very effective for the on-line PN control. Nevertheless, they are valid only with a specific classes of PN. Other approaches, which give generally nonmaximally permissive solution, have been proposed in the literature (Giua et al., 1992; Moody and Antsaklis, 2000; Basile et al., 2006). They take the form of PN which simplifies the analysis and the implementation of the controller. Based on the theory of regions many works have been developed (Ghaffari et al., 2002; Ghaffari et al., 2003; Achour and Rezg, 2006; Lee et al., 2006). The objective is to solve optimally the supervisory control problem using formal and algebraic characterisations. Finally, the PN controller synthesis problem has been treated as integer linear programming problem (Giua and Xie, 2004; Basile et al., 2007a; Basile et al., 2007b). Nevertheless, these approaches require a very high time computing. Although the synthesis problem of PN controller has been widely treated in the literature, the existing approaches cannot be exploited directly in our case because, it is impossible to characterise the forbidden states by constraints. Indeed, the control specification is mainly important to synthesise a controller. However, we have neither the control specification nor the system description because the identified

Bekrar R., Messai N., Essounbouli N., Hamzaoui A. and Riera B. (2008).
SYNTHESIS METHOD OF A PN CONTROLLER USING FORBIDDEN TRANSITIONS SEQUENCES.
In *Proceedings of the Fifth International Conference on Informatics in Control, Automation and Robotics - SPSMC*, pages 149-154
DOI: 10.5220/0001493201490154

PN that we want to control is obtained using only the measurable inputs and outputs system signals. Let's note that, we can determine the forbidden states uniquely by comparing the observed system's states and states reachable by the identified PN model.

This paper deals with the supervisory control problem of DES characterised by forbidden states and modelled by bounded ordinary PN. The PN herein considered is transitions controllable and contains measurable and non measurable places. The latter represent the non measurable outputs system signals. Indeed, the fact that all transitions are controllable then, the PN reachability graph does not contain dangerous markings. In addition, we suppose that we have not the control specification and the initial marking is the unique marked state. Finally, we consider that the PN reachability graph does not contain deadlock states because the input and the output signals observed during the identification phase represent only the normal behaviours of the considered system.
The proposed PN controller is synthesised using the forbidden transitions sequences deduced from the PN reachability graph. The designed PN controller permits to avoid the occurrence of forbidden markings, generated by the non controlled PN model, and guarantees a set of desired behaviours.

In this paper, we start by presenting the considered FSP and defining the control specifications. Then, we present a procedure to calculate the forbidden transitions sequences from the PN reachability graph and introduce a new algorithm to design the PN controller using these sequences. At last, we illustrate the proposed algorithm by an explicative example.

2 CONTROL SPECIFICATIONS

We consider the basic supervisory control problem to synthesise a PN controller that avoids the occurrence of forbidden states. In this paper, we assume that the PN that we want to control is established using the I/O sequences describing all the system normal behaviours according to the algorithm presented in (Bekrar et al., 2006b). Hence, our objective is to add some control places in order to guarantee that all the behaviours generated by the identified PN model correspond to those generated by the real system.
Note also that, each PN reachable marking $M_i^{'}$ is composed of two parts: $M_i^{'} = \begin{bmatrix} M_{im}^{'} \\ M_{i\bar{m}}^{'} \end{bmatrix}$ where, the first one represents the marking of the measurable places and the second part represents the estimated marking of the non measurable places.

The control specifications to be addressed in this paper are the forbidden states type and the problem herein considered becomes a FSP. We will treat it as a FSTP as proved in (Ghaffari et al., 2002). To solve this problem, we propose an algorithm that consists in: (1) identifying the set of the forbidden markings, (2) determining the set of the equivalent forbidden state transitions, (3) developing a PN controller.

2.1 Identification of Forbidden Markings

Contrary to works proposed to treat the FSP where the forbidden states are defined explicitly by constraints, in our case we must calculate them using the behaviours of the real system and those of the PN model. The behaviours of the considered system are represented by the set of its reachable states called $\mathcal{E}$. Those generated by the PN modelling the system are represented by its reachability graph called $R_G^{'}$. Note that, a marking $M_i^{'} \in R_G^{'}$ reachable by the PN that we want to control is said forbidden marking if its measurable part is not equivalent to any state E_i in $\mathcal{E}$ otherwise, it is a legal marking. Hence, the set of the forbidden markings can be obtained using the following procedure:

Input: $\mathcal{E}$ and $R_G^{'}$.
Output: M_{fr} the set of forbidden markings.
Begin

1. Initialise the set of forbidden markings: $M_{fr} := \emptyset$.
2. For each marking $M_i^{'} \in R_G^{'}$ do:
   (a) If there exists a state $E_i \in \mathcal{E}$ such that $M_{im}^{'} = E_i$ then:
      - $M_i^{'}$ is legal marking.
   (b) Else, $M_i^{'}$ is a forbidden marking.
      - Update the set of forbidden markings: $M_{fr} := M_{fr} \cup M_i^{'}$.

End If.
End For.
End.

Once the set of the forbidden markings is obtained, we should determine the set of the transitions leading to or firing from forbidden markings in order to prevent the firing of these transitions.

2.2 Forbidden State Transitions

A state transition in the reachability graph of the PN that we want to control, which fires from a marking $M_i^{'} \in R_G^{'}$ and leads to a marking $M_j^{'} \in R_G^{'}$ ($M_i^{'}[t_i > M_j^{'}$), is said forbidden state transition if and only if one of the following conditions is verified:

- (a) $M_i^{'}$ is forbidden marking or,
- (b) $M_j^{'}$ is forbidden marking.

These can be reformulated, using $\mathcal{E}$ and $R_G^{'}$, as follows: a state transition ($M_i^{'} \xrightarrow{t_i} M_j^{'}$) in $R_G^{'}$ is said forbidden iff:

- $\exists E_i \in \mathcal{E}$: $M_{im}^{'} = E_i$ and $\nexists E_j \in \mathcal{E} : M_{jm}^{'} = E_j$ or,
- $\nexists E_i \in \mathcal{E}$: $M_{im}^{'} = E_i$ and $\exists E_j \in \mathcal{E} : M_{jm}^{'} = E_j$ or,
- $\nexists E_i, E_j \in \mathcal{E} : M_{im}^{'} = E_i$ and $M_{jm}^{'} = E_j$.

Proof: the occurrence of forbidden marking can be prevented by avoiding the firing of the transition leading to this marking. Moreover, it is clear that a transition firing from a forbidden marking is a forbidden transition. So, each transition leading to or firing from a forbidden marking is considered as a forbidden transition.
Therefore, the forbidden state transitions set can be obtained using the following procedure:

Input: $\mathcal{E}$, $R_G^{'}$ and M_{fr}.
Output: Ψ the set of forbidden state transitions.
Begin

1. Initialise the set of forbidden state transitions: $\Psi = \emptyset$.
2. For each state transition ($M_i^{'} \xrightarrow{t_i} M_j^{'}$) $\in R_G^{'}$ such that $M_i^{'}, M_j^{'} \in R_G^{'}$ do:
   - If (a) or (b) is verified then: ($M_i^{'} \xrightarrow{t_i} M_j^{'}$) is a forbidden state transition.
     - Update Ψ:($\Psi := \Psi \cup \{(M_i^{'} \xrightarrow{t_i} M_j^{'})\}$).
   - Else, ($M_i^{'} \xrightarrow{t_i} M_j^{'}$) is a legal state transition.

End If.
End For
End.

Once the forbidden state transitions are determined, the forbidden transitions sequences will be calculated and used for designing a PN controller. More details concerning these steps will be presented in the next section.

3 THE PROPOSED PN CONTROLLER

We can solve the forbidden state problem considered in this paper by adding control places $\{p_{c1}, ..., p_{ck}\}$ to the initial PN model $(N, M_0^{'})$. These places are defined as follows:
Definition 1: A control place p_{ci} of a PN model $(N, M_0^{'})$ is defined by: (i) $M_0^{'}(p_{ci})$: its initial marking, (ii) $Post(p_{ci}, .)$ and $Pre(p_{ci}, .)$: the weighting vectors of the arcs connecting the transitions of $(N, M_0^{'})$ to p_{ci} and connecting p_{ci} to the transitions of $(N, M_0^{'})$ respectively.

Remark 1: Since the considered PN is ordinary then, the arcs weighting values are equal to 1.
In order to solve this problem, we propose to use the forbidden transitions sequences, calculated from the reachability graph of the PN that we want to control, to determine the control places to be added. Note that, these sequences have been used by Lee et al., (2006) but with the constraint asynchronous reachability graph.

3.1 The Forbidden Transitions Sequences

Before introducing the computing procedure of the forbidden transitions sequences, let us present the following definitions that will be used in the PN controller design algorithm.
Definition 2: A transitions sequence σ_i is said forbidden if and only if it allows the firing of at least one forbidden transition. Thus, $\sigma_i = t_1, t_2, ..., t_k, t_l, ..., t_f$ is said forbidden iff: $\exists t_i \in \sigma_i :^{*} M(t_i) \in M_{fr}$ or $M^{*}(t_i) \in M_{fr}$ where $^{*}M(t_i)$ and $M^{*}(t_i)$ represent respectively the input and the output marking of t_i in $R_G^{'}$, otherwise, it is a legal.
The forbidden transitions sequences are calculated using the PN reachability graph as follows:

Input: $R_G^{'}$, Ψ.
Output: S the set of forbidden transitions sequences.
Begin

1. Initialise the set of forbidden transitions sequences: $S := \emptyset$.
2. Calculate $S^{'}$ the set of all legal transitions sequences reachable from the initial marking $M_0^{'}$.
3. Calculate the set of the successors transitions of each transitions sequence σ_i in $S^{'}$, noted $Suc(\sigma_i)$.
4. For each transition t_i in the set $Suc(\sigma_i)$ do:

(a) If t_i is a forbidden transition then:
   - $\sigma_j := \sigma_i t_i$ is a forbidden transitions sequence.
   - Else, stop the successors computation procedure of the transitions sequence σ_j.

(b) Update S ($S := S \cup \{\sigma_j\}$).

End If.
End For

5. Calculate the set of successors transitions of each transitions sequence σ_i in S.
6. Go to 4.
7. End.

Note that, in our case, each forbidden transitions sequence σ_i is composed of two sub-sequences as follows: $\sigma_i = \underbrace{t_1,...,t_k}_{\sigma_i^{''}}|\underbrace{t_l,...,t_f}_{\sigma_i^{in}}$, with $\sigma_i^{''}$ is a legal transitions sub-sequence authorised to be fired from the initial marking $M_0^{'}$ and it can be equal to the empty string ε (i.e., $\sigma_i^{''} = \varepsilon$). Thus, it means that there exists forbidden states reachable from $M_0^{'}$ after the firing of only one transition. σ_i^{in} is a sub-sequence forbidden to firing from $M_k^{'}$ where $M_k^{'}$ is the marking reachable after the firing of t_k. This sub-sequence represents the transitions influenced by the firing of the forbidden transition t_l. We note by $^*\sigma_i^{in}$ and σ_i^{in*} the first and the last transition of σ_i^{in} respectively. Hence, all the markings reachable from $M_0^{'}$ after the firing of the sub-sequence $\sigma_i^{''}$, transition after transition, are legal markings. However, the markings reachable from $M_k^{'}$ after the firing of the sub-sequence σ_i^{in}, transition after transition, are forbidden markings as represented in figure 1.

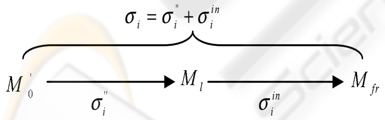

Figure 1: Legal and forbidden markings.

Definition 3: Let's $\sigma_i = t_1, t_2, ..., t_k$ transitions sequence. $|\sigma_i|$ designs the length of this sequence, and it represents the number of transitions in this sequence.

Remark 2: For each transition $t_i \in \sigma_i$, we note by $|t_i \cap \sigma_i|$ the number of times that the transition t_i appears in σ_i.
Definition 4: Let's $\sigma_i = t_1, t_2, ..., t_k$ a transitions sequence that we consider as a word generated by a PN. The prefix of σ_i, noted $Pref(\sigma_i)$, is a word $t_1...t_j$ with $0 \leq j \leq k$ and $k = |\sigma_i|$. For $q \in \mathbb{N}$: $Pref(\sigma_i)^{(q)}$ represents a prefix of σ_i of width equal to q.

Remark 3: Since PN to be controlled is bounded, then its reachability set is finite. Also, the language generated by this PN is finite prefix-closed language.

3.2 Synthesis Algorithm of PN Controller

Based on previous definitions and notations, we propose an algorithm that allows to add control places to the initial PN model for preventing the reachability of forbidden markings. This algorithm contains the following:

1. Construction of the reachability graph of the PN that we want to control.
2. Identification of the forbidden markings and the determination of the forbidden state transitions by browsing the PN reachability graph.
3. Determination of the forbidden transitions sequences set. These latter, we consider them as forbidden words generated by the PN model. Then, we use these sequences to synthesise a PN controller.

Hence, the PN controller will be synthesised using the following algorithm:

Input: $\mathcal{E}$ the system states set, $(N, M_0^{'})$ the PN model.
Output: Controlled PN.
Begin

1. Construct the reachability graph $R_G^{'}$ of the PN model to be controlled $(N, M_0^{'})$.
2. Identify the set of forbidden markings noted M_{fr} as introduced in subsection 2.1.
3. Determine the set of forbidden state transitions Ψ by executing the procedure described in subsection 2.2.
4. Determine the set S of all forbidden transitions sequences firing from the initial marking $M_0^{'}$ by browsing the reachability graph of PN that we want to control as shown in subsection 3.1.
5. Update the set of forbidden transitions sequences S as follows:

(a) For each transitions sequence $\sigma_i \in S$ do:
   i. Calculate the prefix set of this sequence noted $Pref(\sigma_i)$.
   ii. Compare the prefixes of σ_i to the prefixes of all the transitions sequences in S as follows:

A. If there exists at least one transitions sequence σ_j such that, all the elements of prefixes set of σ_i are included in the prefixes set of σ_j ($Pref(\sigma_i) \subset Pref()\sigma_j$) then:
   - Eliminate $\sigma_i$from S: $S := S \backslash \{\sigma_i\}$

B. Else, σ_i still in S.

End If
End For.

6. For each forbidden transitions sequence in S do:
   (a) Determine the sub-sequence of legal transitions $\sigma_i^{''}$ and the sub-sequence of forbidden transitions σ_i^{in}.
   (b) For each sub-sequence of forbidden transitions σ_i^{in}, we determine $^*\sigma_i^{in}$ and σ_i^{in*}.
   (c) Add a control place p_{ci} as an input of $^*\sigma_i^{in}$ and as an output of σ_i^{in*}.
   (d) Mark p_{ci} with initial marking $M_{c0}(p_{ci}) = |^*\sigma_i^{in} \cap \sigma_i^{''}|$. Il there exists several forbidden transitions sub-sequences which have the same $^*\sigma_i^{in}$ and σ_i^{in*} then, we add one control place p_{ci} and we mark it with initial marking equal to $\max_{i,\sigma_i \in S}(|^*\sigma_i^{in} \cap \sigma_i^{''}|)$.

End For.
End.

This algorithm allows to complete the PN model, established using the identification approach proposed previously, by adding control places. The analysis of the algorithm computational complexity shows that it is linear with the number of places, of transitions and, the length of the largest forbidden transitions sequences.

4 ILLUSTRATIVE EXAMPLE

To illustrate the proposed algorithm, let us consider the example of a system described by the identified PN model of figure 2:

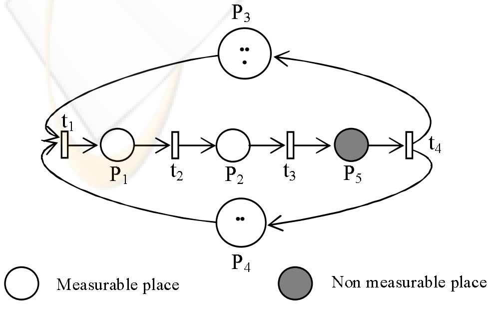

Figure 2: The PN model.

The set of its states is given by: $\mathcal{E} = \left\{ \underbrace{\begin{pmatrix} 0\\0\\3\\2 \end{pmatrix}}_{E_0}, \underbrace{\begin{pmatrix} 1\\0\\2\\1 \end{pmatrix}}_{E_1}, \underbrace{\begin{pmatrix} 0\\1\\2\\1 \end{pmatrix}}_{E_2}, \underbrace{\begin{pmatrix} 0\\0\\2\\1 \end{pmatrix}}_{E_3}, \underbrace{\begin{pmatrix} 2\\0\\1\\0 \end{pmatrix}}_{E_4}, \underbrace{\begin{pmatrix} 1\\1\\1\\0 \end{pmatrix}}_{E_5}, \underbrace{\begin{pmatrix} 1\\0\\1\\0 \end{pmatrix}}_{E_6} \right\}$.

Firstly, we elaborate the PN reachability graph that is given in figure 3.

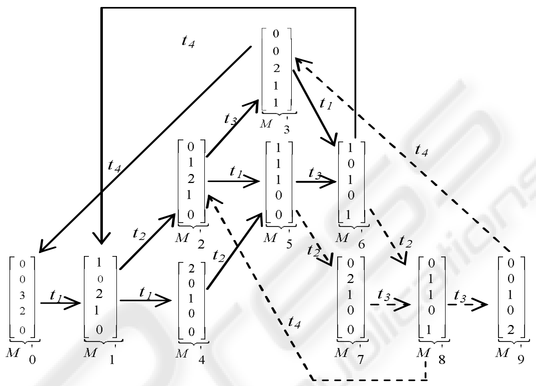

Figure 3: The PN reachability graph.

The set of forbidden markings is: $M_{fr} = \{M_7^{'}, M_8^{'}, M_9^{'}\}$. These markings must be removed from $R_G^{'}$ together with their arcs coming from other markings to these forbidden markings or going from these forbidden markings to other markings in $R_G^{'}$. The set of the equivalent forbidden state transitions are: $\{(M_5^{'} \xrightarrow{t_2} M_7^{'}),\ (M_7^{'} \xrightarrow{t_3} M_8^{'}),\ (M_8^{'} \xrightarrow{t_2} M_9^{'}),\ (M_9^{'} \xrightarrow{t_4} M_3^{'}),\ (M_6^{'} \xrightarrow{t_2} M_8^{'}),\ (M_8^{'} \xrightarrow{t_4} M_2^{'})\}$. By applying the procedure described in subsection 4.1, the forbidden transitions sequences firing from the initial marking are: $\sigma_1 = t_1t_2t_3t_1t_2$, $\sigma_2 = t_1t_2t_3t_1t_2t_4$, $\sigma_3 = t_1t_2t_3t_1t_2t_3$, $\sigma_4 = t_1t_2t_3t_1t_2t_3t_4$, $\sigma_5 = t_1t_1t_2t_3t_2$, $\sigma_6 = t_1t_1t_2t_3t_2t_4$, $\sigma_7 = t_1t_1t_2t_3t_2t_3$, $\sigma_8 = t_1t_1t_2t_3t_2t_3t_4$, $\sigma_9 = t_1t_1t_2t_2$, $\sigma_{10} = t_1t_1t_2t_2t_3$, $\sigma_{11} = t_1t_1t_2t_2t_3t_4$, $\sigma_{12} = t_1t_1t_2t_2t_3t_3$, $\sigma_{13} = t_1t_1t_2t_2t_3t_3t_4$. Thus, the set of the forbidden transitions sequences is $S = \{\sigma_{i,i=1,\ldots,13}\}$. To update S, we must calculate the prefix of each sequence in S. We take as example the sequences σ_1 and σ_2: $Pref(\sigma_1) = \{\varepsilon, t_1, t_1t_2, t_1t_2t_3, t_1t_2t_3t_1, t_1t_2t_3t_1t_2\}$, $Pref(\sigma_2) = \{\varepsilon, t_1, t_1t_2, t_1t_2t_3, t_1t_2t_3t_1, t_1t_2t_3t_1t_2, t_1t_2t_3t_1t_2t_4\}$. We note that $Pref(\sigma_1) = Pref(\sigma_2)$ then, we eliminate σ_1 from S: $S := S \backslash \{\sigma_i\}$. By repeating the same process with the remaining sequences, and after updating S, we have finally : $S = \{\sigma_2, \sigma_4, \sigma_6, \sigma_8, \sigma_{11}, \sigma_{13}\}$. Then, for each forbidden transitions sequence in S we determine the legal transitions sub-sequence and the forbidden one. We take for example $\sigma_8 = \underbrace{t_1t_1t_2t_3}_{\sigma_8^{''}} | \underbrace{t_2t_3t_4}_{\sigma_8^{in}}$. We remark that $\sigma_8^{''} = t_2$ and

$\sigma_8^{in} = t_4$. Finally, by analysing the remaining forbidden sequences, we see that all the forbidden transitions sub-sequences have the same $^*\sigma_i^{in} = t_2$ and $\sigma_i^{in*} = t_4$ for $i \in \{2,4,6,11,13\}$. Therefore, we add one control place p_{c1} with initial marking $M_{oc}(p_{c1}) = \max(|^*\sigma_i^{in} \cap \sigma_i^{''}|) = 1$. This place is an input of t_2 and an output of t_4 as depicted in the figure 4.

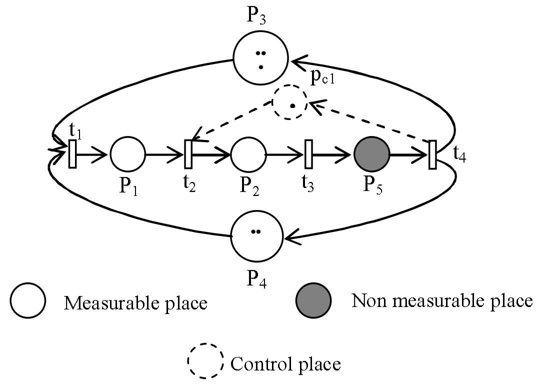

Figure 4: The Controlled PN model.

5 CONCLUSIONS

This paper presents a synthesis method of a PN controller to solve a FSP of DES modelled by a bounded ordinary PN. The model to be controlled is transitions controllable. Using the system behaviours and those generated by the considered PN model, forbidden markings are identified and the equivalent forbidden state transitions are determined. Then, the forbidden transitions sequences deduced from the PN reachability graph are used to synthesise a PN controller. The latter is maximally permissive within the specifications that guarantees the desired behaviours.
As Future work, we will generalise this method to PN with uncontrollable transitions.

REFERENCES

Achour, Z. and Rezg, N. (2006). Time floating general mutual exclusion constraints (tfgmec). In *IMACS, Multiconference on Computational Engineering in Systems Applications*.

Basile, F., Carbone, C., and Chiacchio, P. (2007a). Feedback control logic for backward conflict free choice nets. In *IEEE Transaction on Automation and Control*.

Basile, F., Chiacchio, P., and Giua, A. (2006). Suboptimal supervisory control of Petri nets in presence of uncontrollable transitions via monitor places. In *Automatica*.

Basile, F., P.Chiacchio, and Giua, A. (2007b). An optimisation approach to Petri net monitor design. In *IEEE Transaction on Automation and Control*.

Bekrar, R., Messai, N., Essounbouli, N., Hamzaoui, A., and Riera, B. (2006a). Identification of discrete event systems using ordinary Petri nets. In *IAR-ACD'06, Proceeding of 2006 Workshop on Advanced Control and Diagnosis*.

Bekrar, R., Messai, N., Essounbouli, N., Hamzaoui, A., and Riera, B. (2006b). Off-line identification for a class of discrete event systems using safe Petri nets. In *DES-Des'06, 3rd IFAC Workshop on Discrete Event System Design*.

Boel, R., Ben-Naoum, L., and Breusegem, V. V. (1995). On forbidden state problems for a class of controlled Petri nets. In *IEEE Transaction on Automatic and Control*.

Ghaffari, A., Rezg, N., and Xie, X. (2002). Algebraic and geometric characterization of Petri net controllers using the theory of regions. In *WODES'02, Proceedings of* 6^{th} *Worksop on Discrete Event Systems*.

Ghaffari, A., Rezg, N., and Xie, X. (2003). Feedback control logic for forbidden state problems of marked graphs: application to a real manufacturing system. In *IEEE transaction on Automatic and Control*.

Giua, A., DiCesare, F., and Silva, M. (1992). Generalized mutual exclusion constraints on nets with uncontrollable transitions. In *Proceedings of IEEE International Conference on System, Man and Cybernitic*.

Giua, A. and Xie, X. (2004). Control of safe ordinary Petri nets with marking specifications using unfolding. In *WODES'04, Proceedings of the* 7^{th} *IFAC Workshop on Discrete Event Systems*.

Holloway, L., Guan, X., and Zhang, L. (1996). A generalization of state avoidance policies for controlled Petri nets. In *IEEE Transaction on Automatic and Control*.

Krogh, B. H. and Holloway, L. E. (1991). Synthesis of feedback control logic for discrete manufacturing systems. In *Automatica*.

Lee, E. J., Toguyeni, A., and Dangoumau, N. (2006). A Petri net based approach for the synthesis of parts' controllers for reconfigurable manufacturing systems. In *Proceeding of SICE-ICASE International Joint Conference*.

Moody, J. and Antsaklis, P. (2000). Petri net supervisors for dbs with uncontrollable and unobservable transitions. In *Proceedings of IEEE International Conference on System, Man and Cybernitic*.