CEAI, Vol.11, No.2, pp. 43-50, 2009 Printed in Romania

Control Synthesis for Manufacturing Systems Using Non-Safe Petri Nets

Vasiliu A. I.*. Dideban A.. Alla H.***

*Laboratoire Grenoble Image Parole Signal Automatique (GIPSA Lab), FRANCE
**Semnan University, IRAN

Abstract: This paper addresses the problem of forbidden states in both safe and non-safe Petri net (PN) models. An efficient control synthesis method is presented. Logical control conditions or predicates are associated to controllable transitions with intent to prevent the model from reaching any forbidden states. A simplification technique using over-states is then employed to minimize the final controls. Finally, the problem of control optimality is raised and answered.

Keywords: Discrete Event Systems, Petri Nets, Supervisory Control, Controller Synthesis, Forbidden States

1. INTRODUCTION

The ever-increasing complexity of real-life discrete event systems (DES) brings into focus the need for efficient and realistic control systems in this field. However, although important steps have been made towards the development of such a controller, the problem of complexity still remains a challenging one, especially since it affects the real-time computation time.

The controller task consists of ensuring the fact that the closed loop behaviour of the plant model agrees with the desired specification. Ramadge and Wonham [Ramadge et al. 1987 a, b] have set the ground research in DES control synthesis by proving the existence and portraying the behaviour of a maximally permissive control. Unfortunately, the finite state machine and formal language modelling framework at the basis of this approach makes it very susceptible to the problem of combinatory state explosion, thus rendering it unsuitable for real-life-sized systems.

Petri net (PN) models are to be preferred when dealing with large-scale systems. Apart from a net increase in scalability, this more general DES modelling framework also offers richer modelling structures and implementation alternatives. For this reasons, various attempts have been made to solve the DES control synthesis problem using PN modelling. The algorithm presented in [Li et al. 1994] computes the optimal solution for nets observing the condition of loop-freedom on their uncontrollable sub-net. Uzam and Wonham have developed a hybrid method of DES control synthesis, by coupling automaton supervisors to processes modelled by PNs [Uzam et al. 2006]. This approach has the advantage of being applicable to any type of PN model, although the models' size is considerably increased.

The theory of regions may also be used to create a maximally permissive controller [Ghaffari et al. 2003 a, b]. In this case, however, the controller complexity is increased, as the number of control places is equal to the number of forbidden states.

Yamalidou et al. [Yamalidou et al. 1996] have developed a computationally efficient control synthesis method based on place invariants. The clarity of this method has rendered it very popular. Unfortunately, the resulting controller can no longer ensure the observance of the constraints when the model-controller synchronization is made via uncontrollable transitions. Some simplification techniques based on this method are presented in [Dideban et al. 2005, 2008].

The feedback control synthesis method presented in [Holloway et al. 1990, 1996] has the great advantage of having a large applicability range. Firstly developed for the special case of cyclic controlled marked graphs (CMGs), the method prevents the access to the forbidden states by building conditions that regulate the firing of controllable transitions. However, the control conditions computed this way are often complex. Thus, simplification techniques must be applied, as shown in [Kumar et al, 1996].

A different approach to feedback control is presented in [Dideban et al. 2006]. Contrary to Holloway's approach, this method is based on the analysis of the network reachability graph. This paper gives a detailed presentation of this technique and of its application. The research led by A. Dideban on safe PN models, and presented in detail in [Dideban et al. 2005, 2006, 2008], is complemented by a study of the method's viability in the case of non-safe models. The importance of this study is self-evident, as most real-life systems are characterized by non-safe models. Formal proof for the results stated here for the case of non-safe models may be found in [Vasiliu 2008].

Section 2 of this paper gives the theoretical background behind this approach, starting with a brief presentation of the Holloway & Krogh method, and continuing with a slightly more detailed display of the research led for the case of safe PNs. The particularities of the control synthesis and simplification methods used in the case of non-safe PNs are illustrated in detail in Section 3, while Section 4 demonstrates the method's applicability via a simple example. The conclusions are given in Section 5.

2. THEORETICAL BACKGROUND

The method developed in [Holloway et al. 1990] ensures the synthesis of maximally permissive closed-loop controllers for systems modelled as cyclic CMGs. By restricting the execution of certain controllable events to a number of allowed situations, the system is prevented from ever reaching the forbidden states. The authors make use of the CMG model structure in order to create the controller without generating or analysing the system state-space. Particular attention is given to the effect the control has over the state-place markings of places belonging to *influence paths* (paths beginning with places that are outputs of controlled transitions). The method firstly identifies the sets of places influenced by each control, and then synthesises control conditions capable of forbidding any place belonging to its influence path. Three categories of forbidding conditions are defined, as per their applicability range: *place-*, *set-* and *class- conditions*. Out of these, the latter is also the most general, since any forbidding condition may be written as a class condition.

The same research group later extended this method, so that its applicability range is expanded to a very large class of PNs, including non-safe PNs. The results were published in [Holloway et al. 1996].

The main disadvantage of the Holloway & Krogh method of feedback control for PN models is the growing complexity of the logical expressions associated with control transitions. This in turn leads to the growth of the computation time for the real-time controller. A solution to this problem is given in [Dideban et al. 2006]. A streamlined control condition is obtained by applying a series of simplification techniques to the system state-space.

2.1 Controller synthesis

The first step is to build the reachability graph (off-line) and to identify the controllable transitions, as well as the border-forbidden states and the authorized states. Each controllable transition is then analyzed and its state-space is processed to obtain a control condition that will prevent the system from attaining any forbidden state. The resulting controller is then simplified using properties of the over-state concept.

We shall illustrate the functionality of the synthesis algorithm over the following example:

Example 1: Let us consider a manufacturing system comprising two machines and one robot. The processing capacity of each machine is of one part at a time. Once a machine has finished processing its part (uncontrollable event f_i), the robot unloads it. Once the unloading is completed (uncontrollable event d_i), the robot transfers the processed part into a buffer and then returns to its initial (waiting) state. (The end of the transfer is signalled by the apparition of the uncontrollable event r.) Only the beginning of each processing task is controllable (event c_i).

The system schema, the PN model of its desired closed loop performance and the corresponding reachability graph are given in Figures 1, 2 and 3.

The corresponding sets of authorized- and border-forbidden-states can easily be identified on the reachability graph:

- $M_A = \{P_1P_4P_7, P_2P_4P_7, P_3P_4, P_1P_4P_8, P_1P_5P_7, P_1P_6\}$
- $M_B = \{P_2P_4P_8, P_2P_5P_7, P_3P_5, P_1P_5P_8, P_2P_5P_8, P_2P_6\}$

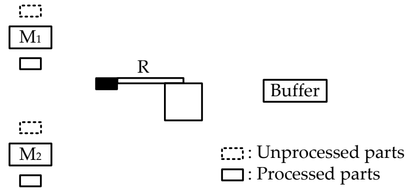

Fig. 1. System schema.

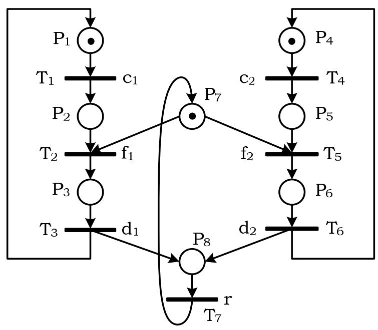

Fig. 2. PN model of the desired closed loop performance.

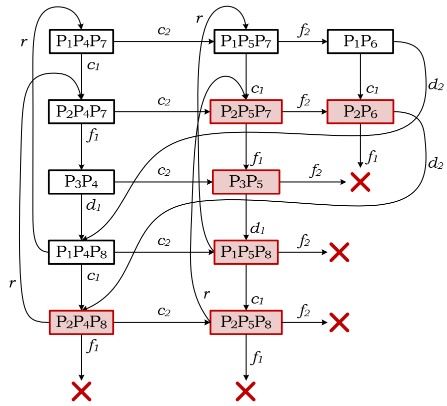

Fig. 3. Reachability graph.

Each controllable transition T_i is associated with two sets of states: a set of *critical states* (M_{Ti}^C) and a set of *safe states* (M_{Ti}^S). Let T_i be a controllable transition, and let M_j be a state from which T_i may fire. M_j is called a *critical state of* T_i if the firing of T_i from M_j leads to a border-forbidden state. Otherwise, M_j is called a *safe state of* T_i. The sets of safe- and critical- states are the basis on which the control condition is to be calculated.

In the case illustrated by *Example 1*, we have two controllable transitions: T_1 and T_4. Their corresponding sets of critical- and safe- states are as follows:

- $M_{T1}^C = \{P_1P_4P_8, P_1P_5P_7, P_1P_6\}$
- $M_{T1}^S = \{P_1P_4P_7\}$

- $M_{T4}^{C} = \{P_1P_4P_8, P_2P_4P_7, P_3P_4\}$
- $M_{T4}^{S} = \{P_1P_4P_7\}$

The control condition can be defined on one hand as a *forbidding condition*; it forbids the firing of its associated transition in any critical state. On the other hand, as a *firing condition*, it allows the firing of its associated transition only from a safe state. Considering the safe PN hypothesis, the forbidding condition for a given critical state is calculated as the logical complement of the intersection of all the place-markings of the state in question.

Let $M_j = P_{j1}P_{j2}...P_{jn}$ be a critical state of controllable transition T_i, and let m_{jr} be the marking of place P_{jr} of M_j. Then the forbidding condition for M_j is given as follows:

$$U_{T_i}\left(M_j\right) = \left(\bigcap_{r=1}^{Card\left[\mathrm{Support}\left(M_j\right)\right]} m_{jr}\right)' \qquad (1)$$

where the support function *Support(M_j)* is defined as the set of marked places of the marking which characterises state M_j.

For example, the forbidding condition for state M = $P_2P_4P_8$ is:

$$U_{T_1}\left(M\right) = \left(\bigcap_{r=1}^{Card\left[\mathrm{Support}\left(M\right)\right]} m_r\right)' = \left(m_2 \cdot m_4 \cdot m_8\right)'$$

The forbidding condition associated to a controllable transition T_i can then be defined as the union of all the forbidding conditions associated to the critical states of T_i.

$$U_{T_i}\left(M_{T_i}^{C}\right) = \bigcup_{j=1}^{card\left[M_{T_i}^{C}\right]} \left(\bigcap_{r=1}^{Card\left[\mathrm{Support}\left(M_j\right)\right]} m_r\right)' \qquad (2)$$

where $m_r \in \mathrm{Support}\left(M_j\right)$ and $M_j \in M_{T_i}^{C}$.

The condition associated to T_1 can therefore be given as:

$$U_{T_1}\left(M_{T_1}^{C}\right) = \bigcup_{j=1}^{card\left[M_{T_i}^{C}\right]} \left(\bigcap_{r=1}^{Card\left[\mathrm{Support}\left(M_j\right)\right]} m_r\right) = \left(\left(m_1 \cdot m_4 \cdot m_8\right) + \left(m_1 \cdot m_5 \cdot m_7\right) + \left(m_1 \cdot m_6\right)\right)'$$

The forbidding condition calculated in this form can still be complex, especially if T_i has a great number of critical states. In answer to this problem, a dual simplification technique is proposed. This technique follows the same basic principles as the simplification of logical expressions, with the added advantages of using the over-state concept.

2.2 Simplification of the control conditions

Let $M_j = P_{j1}P_{j2}...P_{jm}$ be a reachable state. A state $M_i = P_{i1}P_{i2}...P_{in}$ is called *over-state of* M_j if $M_i \le M_j$, or, in terms of support functions: $Support(M_i) \subseteq Support(M_j)$. The control condition associated with an over-state M_i has the remarkable property of simultaneously controlling the access towards all the states covered by M_i. The forbidding condition pertaining to a given collection of over-states, $C_i = \{M_1, M_2, ..., M_m\}$, is defined as follows:

$$U_{T_i}\left(M_{T_i}^{C}\right) = \bigcup_{j=1}^{card\left[C_i\right]} \left(\bigcap_{r=1}^{Card\left[\mathrm{Support}\left(M_j\right)\right]} m_r\right)' \qquad (3)$$

where $m_r \in Support(M_j)$ and $M_j \in C_i$.

The sets of over-states for the T_i critical- and T_i safe- state collections of *Example 1*, i ∈ {1, 4}, are as follows:

- $C_1^{T1} = \{P_1, P_4, P_8, P_1P_4, P_1P_8, P_4P_8, P_1P_4P_8, P_5, P_7, P_1P_5, P_1P_7, P_5P_7, P_1P_5P_7, P_6, P_1P_6\}$
- $S_1^{T1} = \{P_1, P_4, P_7, P_1P_4, P_1P_7, P_4P_7, P_1P_4P_7\}$
- $C_1^{T4} = \{P_1, P_4, P_8, P_1P_4, P_1P_8, P_4P_8, P_1P_4P_8, P_2, P_7, P_2P_4, P_2P_7, P_4P_7, P_2P_4P_7, P_3, P_3P_4\}$
- $S_1^{T4} = \{P_1, P_4, P_7, P_1P_4, P_1P_7, P_4P_7, P_1P_4P_7\}$

An over-state may cover both critical and safe states. In order to ensure control optimality, a simplified control condition may only be used if it does not hinder the reachability of any safe states. Let T_i be a controllable transition, and let C_1^{Ti} and S_1^{Ti} be the collections of over-states corresponding, respectively, to its sets of critical- and safe- states. The simplified forbidding condition for T_i relates to the collection of over-states defined by:

$$C_2^{T_i}\left(M_j\right) = C_1^{T_i} \setminus S_1^{T_i} \qquad (4)$$

The ensuing collection of over-states must then be filtered of redundancies.

Below are given the sets of control over-states corresponding to the two controllable transitions in *Example 1*:

- $C_2^{T1} = C_1^{T1} \setminus S_1^{T1} = \{P_8, P_1P_8, P_4P_8, P_1P_4P_8, P_5, P_1P_5, P_5P_7, P_1P_5P_7, P_6, P_1P_6\}$
- $C_2^{T4} = C_1^{T4} \setminus S_1^{T4} = \{P_8, P_1P_8, P_4P_8, P_1P_4P_8, P_2, P_2P_4, P_2P_7, P_2P_4P_7, P_3, P_3P_4\}$

Some over-states in C_2^{Ti}, i ∈ {1, 4}, are covered by other. The final sets of control over-states are obtained after filtering out these redundancies:

- $C_3^{T1} = \{P_8, P_5, P_6\}$
- $C_3^{T4} = \{P_8, P_2, P_3\}$

In order to select the simplest control condition possible, the final set of control over-states must be minimized. The final choice algorithm used is similar to the Clusky method. As a first step, the collection of covered critical states is written for each over-state in C_2^{Ti}. Then the set of over-states corresponding to each critical state is examined, and the over-states that cover in exclusivity one or more critical states are chosen. In the case of the critical states covered by more than one over-state, the over-state covering the maximum of critical states must be chosen.

An intuitive application of the final choice algorithm on the results obtained for *Example 1* is illustrated in the following tables:

Table 1. Final choice for T_1 – forbidding condition

C_3^{T1} \ Critical state	$P_1P_4P_8$	$P_1P_5P_7$	P_1P_6	Choice (C_4^{T1})
P_8	✓			✓
P_5		✓		✓
P_6			✓	✓
	✓	✓	✓	

Table 2. Final choice for T_4 – forbidding condition

C_3^{T4} \ Critical state	$P_1P_4P_8$	$P_2P_4P_7$	P_3P_4	Choice (C_4^{T4})
P_8	✓			✓
P_2		✓		✓
P_3			✓	✓
	✓	✓	✓	

The forbidding conditions can now be calculated:

- $U_{T1}(M_{T1}{}^{C}) = (m_8 + m_5 + m_6)'$
- $U_{T4}(M_{T4}{}^{C}) = (m_8 + m_2 + m_3)'$

So far, only the synthesis of the forbidding condition has been discussed. Assuming M_j were a safe state, the firing control condition could be calculated and simplified in a similar manner, as shown in [Dideban et al. 2006].

The firing condition for *Example 1* is determined below:

Firstly, the sets of firing control over-states are determined:

- $S_2^{T1} = S_1^{T1} \backslash C_1^{T1} = \{P_4P_7, P_1P_4P_7\}$
- $S_2^{T4} = S_1^{T4} \backslash C_1^{T4} = \{P_1P_7, P_1P_4P_7\}$

Then the redundancies are filtered out:

- $S_3^{T1} = \{P_4P_7\}$
- $S_3^{T4} = \{P_1P_7\}$

In this case the final collection of control over-states is already minimal. There is therefore no need to apply the final choice algorithm. The firing conditions for the two controllable transitions are given below:

- $U_{T1}(M_{T1}{}^{C}) = m_4 \cdot m_7$
- $U_{T4}(M_{T4}{}^{C}) = m_1 \cdot m_7$

At this point, the problem of controller optimality must be formally addressed. A maximally permissive controller must simultaneously fulfil these two basic conditions: (1) effectively prevent its associated transition from firing towards any forbidden state, and (2) allow the aforementioned transition to fire towards any authorized state. This happens if and only if the set of reachable states of the controlled model is equal to the set of authorized states of the initial model: $M_{NI} = M_A$. This fact can be proven using the coverage relations governing the associations between the collections of states and over-states pertaining to a certain controllable transition T_i, $R_{Ti}(M_j, c_k)$ and $CV_{Ti}(M_j, C_i)$. The formal proof for this statement is given in [Dideban et al. 2006].

Tables 3 and 4 give the proof of maximal permissiveness for the forbidding controls calculated for *Example 1*. In the case of the firing conditions, a simple visual evaluation of the results is enough to prove maximal permissiveness.

Table 3. Proof of maximal permissiveness for $U_{T1}(M_{T1}{}^{C})$

c_k \ M_j	$P_1P_4P_8$	$P_1P_5P_7$	P_1P_6
P_8	1	0	0
P_5	0	1	0
P_6	0	0	1
$CV_{T1}(M_j, C_3^{T1})$	1	1	1

Table 4. Proof of maximal permissiveness for $U_{T4}(M_{T4}{}^{C})$

c_k \ M_j	$P_1P_4P_8$	$P_2P_4P_7$	P_3P_4
P_8	1	0	0
P_2	0	1	0
P_3	0	0	1
$CV_{T4}(M_j, C_3^{T4})$	1	1	1

2.3 The feedback control synthesis algorithm

Let M_A be the set of authorized states and M_B be the set of border-forbidden states of a given PN model, and let T_i be a controllable transition of this model. The feedback control logic synthesis algorithm is given below:

Algorithm 1:

Step 1: Build the sets of T_i critical- and T_i safe- states ($M_{Ti}{}^{C}$ & $M_{Ti}{}^{S}$)

Step 2: Build the corresponding sets of over-states for $M_{Ti}{}^{C}$ & $M_{Ti}{}^{S}$ (C_1^{Ti} & C_2^{Ti})

Step 3: Build the sets of over-states corresponding to the simplified forbidding- and firing- control conditions (C_2^{Ti} & S_2^{Ti})

Step 4: Filter out the redundancies (C_3^{Ti} & S_3^{Ti})

Step 5: Select the minimal sets of control over-states according to the Clusky method (C_4^{Ti} & S_4^{Ti})

Step 6: Which of these two sets of over-states is maximally permissive?

If both: go to *Step 7*

If one: select it and go to *Step 8*

If none: there is no maximally permissive controller ⇒ STOP

Step 7: Select the simplest controller (C_4^{Ti} or S_4^{Ti})

Step 8: Calculate the controller

The results obtained after applying the synthesis algorithm on the system presented in *Example 1* are given in Table 5:

Table 5. Controller synthesis for *Example 1*

Forbidding conditions:	Firing conditions:
- $U_{T1}(M_{T1}^{C}) = (m_8 + m_5 + m_6)'$ - $U_{T4}(M_{T4}^{C}) = (m_8 + m_2 + m_3)'$	- $U_{T1}(M_{T1}^{C}) = m_4 \cdot m_7$ - $U_{T4}(M_{T4}^{C}) = m_1 \cdot m_7$

While both the firing- and the forbidding- conditions give maximally permissive controllers, the firing conditions are, in this case, simpler.

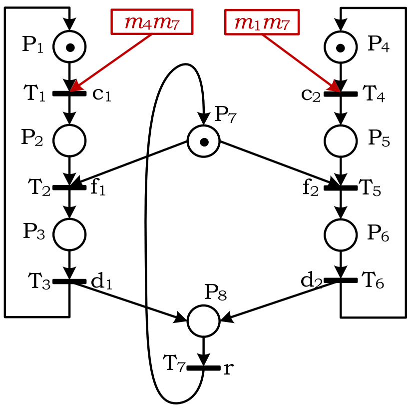

Fig. 4. Final controlled model.

3. FEEDBACK CONTROL OF NON-SAFE PETRI NETS

The applicability range of the feedback control synthesis method presented in [Dideban et al. 2006] can be extended to include non-safe PNs. The basic idea and final synthesis algorithm are the same, but a few modifications must be made in order to accommodate the non-safe PN model structure. The formal proof of the results presented in this section is given in [Vasiliu 2008].

Let $\{0, 1, ..., i\}^N$, $i \in \mathbb{N}$, be the collection of all N-dimensional vectors. The marking of a non-safe PN is a member of $\{0, 1, ..., i\}^N$. The support function of one such marking, M_j, is defined as follows:

$$\text{Support}\left(M_j\right) = \left\{\prod_r P_{jr}^{kr}\right\} \tag{5}$$

where P_{jr} is a marked place of M_j and kr is the number of tokens in P_{jr}.

Let T_i be a controllable transition, M_{Ti}^{C} and M_{Ti}^{S} be its corresponding sets of critical- and safe- states, and M_j be a state reachable by firing T_i. The dual definition of the control condition U_{Ti}: $(M_{Ti}^{C}, M_j) \rightarrow \{0, 1\}$ remains unchanged:

Forbidding condition:

$$U_{T_i}\left(M_{T_i}^{C}, M_j\right) = \begin{cases} 0, & M_j \in M_{T_i}^{C} \\ 1, & M_j \notin M_{T_i}^{C} \end{cases} \tag{6}$$

Firing condition:

$$U_{T_i}\left(M_{T_i}^{C}, M_j\right) = \begin{cases} 0, & M_j \in M_{T_i}^{S} \\ 1, & M_j \notin M_{T_i}^{S} \end{cases}$$

In the case of safe PNs, the binary nature of the place markings allowed for an alternative definition of the control condition; a definition based on the values of the place markings. In order to continue using that definition in the case of non-safe PNs, a way of expressing place markings as binary variables must be found.

Let $M_j = P_{j1}^{k1} P_{j2}^{k2} ... P_{jn}^{kn}$ be a T_i critical state. The *marking predicate* Z_{jr} of place P_{jr} in a given state M_i is defined as follows:

$$Z_{jr} = \begin{cases} 1, & m_i\left(P_{jr}\right) \geq k_{jr} \\ 0, & m_i\left(P_{jr}\right) < k_{jr} \end{cases} \tag{7}$$

The alternative definition for the control condition can therefore be written as follows:

$$U_{T_i}\left(M_{T_i}^{C}\right) = \bigcup_{j=1}^{card\left[M_{T_i}^{C}\right]} \left(\bigcap_{r=1}^{Card\left[\text{Support}\left(M_j\right)\right]} Z_r \right)' \tag{8}$$

In order to simplify this condition, we must first extend the notion of over-state to accommodate the non-safe PN model. Let $M_j = P_{j1}^{k1} P_{j2}^{k2} ... P_{jm}^{km}$ be a reachable state of a non-safe PN model. $M_i = P_{i1}^{k1} P_{i2}^{k2} ... P_{in}^{kn}$ is called an over-state of M_j if: $\forall\, P_{ir} \in Support(M_i)$ and $\forall\, P_{jr} \in Support(M_j) \mid P_{ir} \equiv P_{jr}$ and $k_{ir} \leq k_{jr}$. As in the case of safe PNs, if M_i is over-state of M_j then $Support(M_i) \subseteq Support(M_j)$.

Property 1: Let $M_1 = P_{11}^{k1} P_{12}^{k2} ... P_{1m}^{km} ... P_{1n}^{kn}$ be a T_i critical state, and let $M_2 = P_{11}^{k1} P_{12}^{k2} ... P_{1m}^{km}$ be an over-state of M_1. The control condition associated to M_2 also affects M_1:

$$U_{T_i}\left(M_1\right) = 0 \Rightarrow U_{T_i}\left(M_2\right) = 0 \tag{9}$$

An optimal controller must not forbid any authorized states:

Property 2: Let M_{Ti}^{C} and M_{Ti}^{S} be the sets of T_i critical- and safe- states, and let $M_1 = P_{11}^{k1} P_{12}^{k2} ... P_{1m}^{km} ... P_{1n}^{kn} \in M_{Ti}^{C}$ be a T_i critical state and $M_2 = P_{11}^{k1} P_{12}^{k2} ... P_{1m}^{km}$ be an over-state of M_1. The control condition $U_{Ti}(M_1)$ can be substituted by $U_{Ti}(M_2)$ only if there is no safe state $M_3 \in M_{Ti}^{S}$ so that $M_2 < M_3$.

At this point the forbidding condition pertaining to a given collection of over-states $C_i = \{M_1, M_2, ..., M_m\}$, $M_j = P_{j1}^{k1} P_{j2}^{k2} ... P_{jr}^{kr} ... P_{jn}^{kn} \in M_{Ti}^{C}$, can be defined as follows:

$$U_{T_i}\left(M_{T_i}^{C}\right) = \bigcup_{j=1}^{card\left[C_i\right]} \left(\bigcap_{r=1}^{Card\left[\text{Support}\left(M_j\right)\right]} Z_r \right)' \tag{10}$$

If M_j were a safe state, the firing condition could be calculated similarly:

$$U_{T_i}\left(M_{T_i}^{C}\right)=\bigcup_{j=1}^{card\left[S_i\right]}\left(\bigcap_{r=1}^{Card\left[\text{Support}\left(M_j\right)\right]} Z_r\right)' \tag{11}$$

The minimal set of control over-states is selected using the Clusky method, in the same manner as in the case of safe PNs, and the synthesis procedure follows the same steps.

The proof of controller optimality entails once more the extension of certain notions as to accommodate the non-safe PN model. A state can be covered both by a single over-state and by a set of over-states. The coverage relations that govern these associations can be redefined as follows:

Let c_k be an over-state of the final set of control over-states, $C_3^{Ti} = \{c_1, ..., c_k, ..., c_m\}$, and let M_j be a T_i critical state, $M_j \in M_{Ti}^C$. The *coverage relations* R_{Ti}: $M_{Ti}^C \times C_3^{Ti} \rightarrow \{0, 1\}$ and CV_{Ti}: $M_{Ti}^C \times C_3^{Ti} \rightarrow \{0, 1\}$ are given as follows:

$$R_{T_i}\left(M_j, c_k\right)=\begin{cases}1, & c_k \le c_j\left(c_k \text{ is over-state of } c_j\right)\\ 0, & c_k \quad \text{is not an over-state of } c_j\end{cases}, \tag{12}$$

where c_j is the over-state associated to M_j;

$$CV_{T_i}\left(M_j, C_3^{T_i}\right)=\bigcup_{k=1}^{m} R_{T_i}\left(M_j, c_k\right) \tag{13}$$

The following property states the necessary and sufficient condition for a controller to be maximally permissive:

Property 3: Let T_i be a controllable transition. The set of controls $U_{Ti}(C_3^{Ti}, M_k)$ gives a maximally permissive controller if and only if: $\forall\, M_j \in M_{Ti}^C$: $CV_{Ti}(M_j, C_3^{Ti}) = 1$.

The general steps of the synthesis procedure are those illustrated in *Algorithm 1*.

4. MANUFACTURING SYSTEM

The general controller synthesis algorithm will be illustrated over an extension of *Example 1*. We shall now consider we have two robots. Figures 5 and 6 give the PN model and reachability graph for this system.

The sets of authorized- and border-forbidden- states can easily be identified on the reachability graph:

- $M_A = \{P_1P_4P_7^2, P_2P_4P_7^2, P_3P_4P_7, P_1P_4P_7P_8, P_2P_4P_7P_8, P_3P_4P_8, P_1P_4P_8^2, P_1P_5P_7^2, P_2P_5P_7^2, P_3P_5P_7, P_1P_5P_7P_8, P_1P_6P_7, P_2P_6P_7, P_3P_6, P_1P_6P_8\}$
- $M_B = \{P_2P_4P_8^2, P_2P_5P_7P_8, P_3P_5P_8, P_1P_5P_8^2, P_2P_5P_8^2, P_2P_6P_8\}$

The sets of critical- and safe- states corresponding to the controllable transitions T_1 and T_4 are as follows:

- $M_{T1}^C = \{P_1P_4P_8^2, P_1P_5P_7P_8, P_1P_6P_8\}$
- $M_{T1}^S = \{P_1P_4P_7^2, P_1P_4P_7P_8, P_1P_5P_7^2, P_1P_6P_7\}$
- $M_{T4}^C = \{P_1P_4P_8^2, P_2P_4P_7P_8, P_3P_4P_8\}$
- $M_{T4}^S = \{P_1P_4P_7^2, P_1P_4P_7P_8, P_2P_4P_7^2, P_3P_4P_7\}$

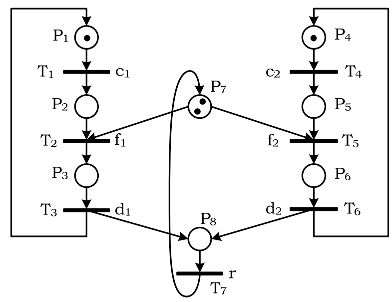

Fig. 5. PN model of the desired closed loop performance.

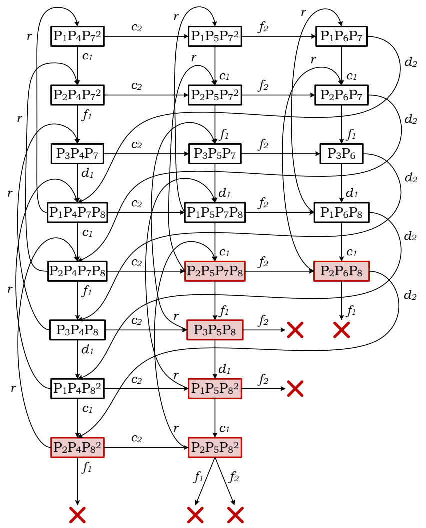

Fig. 6. Reachability graph.

The over-state sets for the T_i critical- and T_i safe- state collections, i ∈ {1, 4}:

- $C_1^{T1} = \{P_1, P_4, P_8, P_8^2, P_1P_4, P_1P_8, P_4P_8, P_1P_8^2, P_4P_8^2, P_1P_4P_8, P_1P_4P_8^2, P_5, P_7, P_1P_5, P_1P_7, P_5P_7, P_5P_8, P_7P_8, P_1P_5P_7, P_1P_5P_8, P_1P_7P_8, P_5P_7P_8, P_1P_5P_7P_8, P_6, P_1P_6, P_6P_8, P_1P_6P_8\}$
- $S_1^{T1} = \{P_1, P_4, P_7, P_7^2, P_1P_4, P_1P_7, P_4P_7, P_1P_7^2, P_4P_7^2, P_1P_4P_7, P_1P_4P_7^2, P_8, P_1P_8, P_4P_8, P_7P_8, P_1P_4P_8, P_1P_7P_8, P_4P_7P_8, P_1P_4P_7P_8, P_5, P_1P_5, P_5P_7, P_5P_7^2, P_1P_5P_7, P_1P_5P_7^2, P_6, P_1P_6, P_6P_7, P_1P_6P_7\}$
- $C_1^{T4} = \{P_1, P_4, P_8, P_8^2, P_1P_4, P_1P_8, P_4P_8, P_1P_8^2, P_4P_8^2, P_1P_4P_8, P_1P_4P_8^2, P_2, P_7, P_2P_4, P_2P_7, P_2P_8, P_4P_7, P_7P_8, P_2P_4P_7, P_2P_4P_8, P_2P_7P_8, P_4P_7P_8, P_2P_4P_7P_8, P_3, P_3P_4, P_3P_8, P_3P_4P_8\}$
- $S_1^{T4} = \{P_1, P_4, P_7, P_7^2, P_1P_4, P_1P_7, P_4P_7, P_1P_7^2, P_4P_7^2, P_1P_4P_7, P_1P_4P_7^2, P_8, P_1P_8, P_4P_8, P_7P_8, P_1P_4P_8, P_1P_7P_8, P_4P_7P_8, P_1P_4P_7P_8, P_2, P_2P_4, P_2P_7, P_2P_7^2, P_2P_4P_7, P_2P_4P_7^2, P_3, P_3P_4, P_3P_7, P_3P_4P_7\}$

The sets of over-states needed to calculate the forbidding conditions corresponding to the two controllable transitions:

- $C_2^{T1} = C_1^{T1} \backslash S_1^{T1} = \{P_8^2, P_1P_8^2, P_4P_8^2, P_1P_4P_8^2, P_5P_8, P_1P_5P_8, P_5P_7P_8, P_1P_5P_7P_8, P_6P_8, P_1P_6P_8\}$
- $C_2^{T4} = C_1^{T4} \backslash S_1^{T4} = \{P_8^2, P_1P_8^2, P_4P_8^2, P_1P_4P_8^2, P_2P_8, P_2P_4P_8, P_2P_7P_8, P_2P_4P_7P_8, P_3P_8, P_3P_4P_8\}$

We notice that certain over-states in C_2^{Ti}, i ∈ {1, 4}, are covered by other. The minimal sets of control over-states are obtained after filtering out the redundancies:

- $C_3^{T1} = \{P_8^2, P_5P_8, P_6P_8\}$
- $C_3^{T4} = \{P_8^2, P_2P_8, P_3P_8\}$

An intuitive application of the final choice algorithm is illustrated in the following tables:

Table 6. Final choice for T_1 – forbidding condition

C_3^{T1} \ Critical state	$P_1P_4P_8^2$	$P_1P_5P_7P_8$	$P_1P_6P_8$	Choice (C_4^{T1})
P_8^2	✓			✓
P_5P_8		✓		✓
P_6P_8			✓	✓
	✓	✓	✓	

Table 7. Final choice for T_4 – forbidding condition

C_3^{T4} \ Critical state	$P_1P_4P_8^2$	$P_2P_4P_7P_8$	$P_3P_4P_8$	Choice (C_4^{T4})
P_8^2	✓			✓
P_2P_8		✓		✓
P_3P_8			✓	✓
	✓	✓	✓	

The forbidding conditions can now be calculated:

- $U_{T1}(M_{T1}^C) = (Z_8 + Z_5 \cdot Z_8 + Z_6 \cdot Z_8)'$
- $U_{T4}(M_{T4}^C) = (Z_8 + Z_2 \cdot Z_8 + Z_3 \cdot Z_8)'$

Table 8. Proof of maximal permissiveness for $U_{T1}(M_{T1}^C)$

c_k \ M_j	$P_1P_4P_8^2$	$P_1P_5P_7P_8$	$P_1P_6P_8$
P_8^2	1	0	0
P_5P_8	0	1	0
P_6P_8	0	0	1
$CV_{T1}(M_j, C_3^{T1})$	1	1	1

The control maximal permissiveness is proven using the coverage relations and *Property 3*, as shown in Tables 8 and 9.

Table 9. Proof of maximal permissiveness for $U_{T4}(M_{T4}^C)$

c_k \ M_j	$P_1P_4P_8^2$	$P_2P_4P_7P_8$	$P_3P_4P_8$
P_8^2	1	0	0
P_2P_8	0	1	0
P_3P_8	0	0	1
$CV_{T4}(M_j, C_3^{T4})$	1	1	1

The firing condition is determined in a similar manner. Firstly, the sets of firing control over-states are determined:

- $S_2^{T1} = S_1^{T1} \backslash C_1^{T1} = \{P_7^2, P_4P_7, P_1P_7^2, P_4P_7^2, P_1P_4P_7, P_1P_4P_7^2, P_4P_7P_8, P_1P_4P_7P_8, P_5P_7^2, P_1P_5P_7^2, P_6P_7, P_1P_6P_7\}$
- $S_2^{T4} = S_1^{T4} \backslash C_1^{T4} = \{P_7^2, P_1P_7, P_1P_7^2, P_4P_7^2, P_1P_4P_7, P_1P_4P_7^2, P_1P_7P_8, P_1P_4P_7P_8, P_2P_7^2, P_2P_4P_7^2, P_3P_7, P_3P_4P_7\}$

Then the redundancies are filtered out:

- $S_3^{T1} = \{P_7^2, P_4P_7, P_6P_7\}$
- $S_3^{T4} = \{P_7^2, P_1P_7, P_3P_7\}$

The final choice algorithm is applied:

Table 10. Final choice for T_1 – firing condition

S_3^{T1} \ Safe state	$P_1P_4P_7^2$	$P_1P_4P_7P_8$	$P_1P_5P_7^2$	$P_1P_6P_7$	Choice (S_4^{T1})
P_7^2	✓		✓		✓
P_4P_7	✓	✓			✓
P_6P_7				✓	✓
	✓	✓	✓	✓	

Table 11. Final choice for T_4 – firing condition

S_3^{T4} \ Safe state	$P_1P_4P_7^2$	$P_1P_4P_7P_8$	$P_2P_4P_7^2$	$P_3P_4P_7$	Choice (S_4^{T4})
P_7^2	✓		✓		✓
P_1P_7	✓	✓			✓
P_3P_7				✓	✓
	✓	✓	✓	✓	

Finally, the firing conditions are calculated for the two controllable transitions:

- $U_{T1}(M_{T1}^C) = Z_7 + Z_4 \cdot Z_7 + Z_6 \cdot Z_7$
- $U_{T4}(M_{T4}^C) = Z_7 + Z_1 \cdot Z_7 + Z_3 \cdot Z_7$

Tables 12 and 13 show that these controls are maximally permissive as well:

Table 12. Proof of maximal permissiveness for $U_{T1}(M_{T1}^C)$

c_k \ M_j	$P_1P_4P_7^2$	$P_1P_4P_7P_8$	$P_1P_5P_7^2$	$P_1P_6P_8$
P_7^2	1	0	1	0
P_4P_7	1	1	0	0
P_6P_7	0	0	0	1
$CV_{T1}(M_j, S_3^{T1})$	1	1	1	1

Table 13. Proof of maximal permissiveness for $U_{T4}(M_{T4}^C)$

c_k \ M_j	$P_1P_4P_7^2$	$P_1P_4P_7P_8$	$P_2P_4P_7^2$	$P_3P_4P_8$
P_7^2	1	0	1	0
P_1P_7	1	1	0	0
P_3P_7	0	0	0	1
$CV_{T4}(M_j, S_3^{T4})$	1	1	1	1

In this case, both the forbidding and the firing control conditions are maximally permissive. The next step is therefore to compare the results and choose the simplest possible solution.

A visual comparison of the two controls obtained for each controllable transition shows no obvious difference in their complexity levels. Thus, either control may be used with similar results.

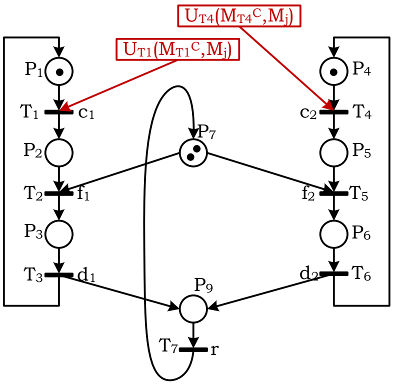

Fig. 7. Final controlled model.

5. CONCLUSIONS

The problem of controller synthesis for DES remains a redoubtable one. Traditional methods based on finite-state-machine models are very sensitive to the combinatory-state-explosion problem, while methods based on simpler PN models are confronted with the problem of controller optimality. In this context, the feedback control synthesis method presented in this paper provides a systematic and easily implementable tool for specialists in this field.

Apart from its simplicity, this method has the great advantage of being applicable to both safe and non-safe PN models. The problem of controller optimality is also addressed, in the sense that, if a maximally permissive controller exists, it will be found amongst the results. Furthermore, the final (controlled) model is close in size to the initial model.

We believe that the controller complexity may be reduced even further by the development of more efficient algorithms for the construction of the control over-states collections.

REFERENCES

Dideban, A., Alla, H. (2005). From Forbidden States to Linear Constraints for the Optimal Supervisory Control. *Control Engineering and Applied Informatics (CEAI)*, 7(3),48 – 55.

Dideban, A., Alla, H. (2006). Solving the Problem of Forbidden States by Feedback Control Logical Synthesis. *The 32nd Annual Conference of the IEEE Industrial Electronics Society (IECON)*, 7 – 10 Nov., Paris, France. Proceedings: 348 – 353.

Dideban, A., Alla, H. (2008). Reduction of Constraints for Controller Synthesis Based on Safe Petri Nets. *Automatica*, 44(7), 1697 – 1706.

Ghaffari, A., Rezg, N., Xie, X. (2003 a). Feedback Control Logic for Forbidden-State Problems of Marked Graphs: Application to a Real Manufacturing System. *IEEE Transactions on Automatic Control*, 48(1), 18 – 29.

Ghaffari, A., Rezg, N., Xie, X. (2003 b). Design of a Live and Maximally Permissive Petri Net Controller Using the Theory of Regions. *IEEE Transactions on Robotics and Automation*, 19(1), 137 – 142.

Holloway, L.E., Krogh, B.H. (1990). Synthesis of Feedback Control Logic for a Class of Controlled Petri Nets. *IEEE Transactions on Automatic Control*, 35(5), 514 – 523.

Holloway, L.E., Guan, X., Zhang, L. (1996). A Generalization of State Avoidance Policies for Controlled Petri Nets. *IEEE Transactions on Automatic Control*, 41(6), 804 – 816.

Kumar, R., Holloway, L.E. (1996). Feedback Control of Deterministic Petri Nets with Regular Specification Languages. *IEEE Transactions on Automatic Control*, 41(2), 245 – 249.

Li, Y., Wonham, W.M. (1994). Control of Vector Discrete Event Systems II: Controller Synthesis. *IEEE Transactions on Automatic Control*, 39(3), 512 – 531.

Ramadge, P.J., Wonham, W.M. (1987 a). Supervisory Control of a Class of Discrete Event Processes. *SIAM Journal of Control and Optimization*, 25(1), 206 – 230.

Ramadge, P.J., Wonham, W.M. (1987 b). On the Supremal Controllable Sublanguage of a Given Language. *SIAM Journal of Control and Optimization*, 25(3), 637 – 659.

Uzam, M., Wonham, W.M. (2006). A Hybrid Approach to Supervisory Control of Discrete Event Systems Coupling RW Supervisors to Petri Nets. *International Journal of Advanced Manufacturing Technologies*, 28, 747 – 760.

Vasiliu, A.I. (2008). Internal Report – Synthèse par retour d'état de contrôleurs discrets pour les systèmes manufacturiers modélises par des réseaux de Petri saufs et non-saufs.

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