Petri Net Supervisors for Generalized Mutual Exclusion Constraints

Alessandro Giua,

Dip. di Ingegneria Elettrica ed Elettronica, Università di Cagliari,

Piazza d'Armi — 09123 Cagliari, Italy

Phone: +39-070-675-5892 – Fax: +39-070-675-5900 – Email: giua@diee.unica.it.

Frank DiCesare

Electrical, Computer, and Systems Eng. Dept., Rensselaer Polytechnic Institute

Troy, NY 12180-3590, USA

Manuel Silva

Dpto. Ing. Eléctrica e Informática, Universidad de Zaragoza

50015, Zaragoza, Spain

Abstract

The paper discusses the problem of enforcing generalized mutual exclusion constraints on place/transition nets with uncontrollable transitions. For a class of Petri nets, marked graphs with control safe places, two Petri net structures capable of enforcing these constraints are presented. The first one, is a monitor-based solution; the second one a supervisory based solution. Both structures are fully compiled, i.e., they are given as simple place/transition nets with no associated predicates, thus permitting the construction and the analysis of a closed loop-model of the controlled system.

Published as:

A. Giua, F. DiCesare, M. Silva, "Petri Net Supervisors for Generalized Mutual Exclusion Constraints", Proc. 12th IFAC World Congress (Sidney, Australia), Vol. 1, pp. 267-270, July 1993.

1 Introduction

Mutual exclusion constraints are a natural way of expressing the concurrent use of a finite number of resources, shared among different processes. For systems represented as Petri nets, Giua *et al.* (1992) have defined a generalized mutual exclusion constraint (GMEC) as a condition that limits a weighted sum of tokens contained in a subset of places.

In traditional Petri net modeling all transitions are assumed to be *controllable*, i.e., may be prevented from firing by a control agent. A single GMEC may be easily implemented by a *monitor*, i.e., a place whose initial marking represents the available units of a resource and whose outgoing and incoming transitions represent, respectively, the acquisition and release of units of the resource.

In the framework of Supervisory Control (Ramadge and Wonham, 1989) the complexity of enforcing a GMEC is enhanced by the presence of *uncontrollable* transitions, i.e., transitions that may be observed but not prevented from firing by a control agent. To enforce a given GMEC, it is necessary to prevent the system from reaching not only the forbidden markings (i.e., those markings that do not satisfy the constraint), but also all those markings from which a forbidden one may be reached by firing a sequence of uncontrollable transitions as discussed in Golaszewski and Ramadge (1988). Unfortunately, in this case it was shown that there exist problems which do not have a "monitor-based" solution (Giua *et al.*, 1992).

In this context, this paper discusses GMEC for systems represented as marked graphs. The goal is that of constructing a supervisor capable of enforcing the constraints. A solution to this problem has been given by Krogh and Holloway (1991). In their approach, which may be defined as *fully interpreted*, the control policy is efficiently computed by an on-line controller as a feedback function of the marking of the system. In this work, instead, a net based structure for the supervisor is built. The paper presents and compares two different solutions that are *fully compiled*, i.e., the corresponding supervisor is represented by a place/transition net.

There are several advantages in fully compiling the supervisor action into a net structure. Firstly, the computation of the control action is faster, since it does not require separate on-line computation. Secondly, the same Petri net system execution algorithms may be used for both the original system and the supervisor. Finally, a closed-loop model of the system under control may be built with standard net composition constructions, and

efficiently analyzed for structural properties of interest.

The proofs of the propositions presented in the paper can be found in Giua (1992).

2 Generalities

A *place/transition net* (P/T net) is a structure $N = (P, T, Pre, Post)$. Readers unfamiliar with the notation may refer to Murata (1989).

The *preset* (*postset*) of a subset of places $P_i \subseteq P$ is the set: ${}^\bullet P_i = \{t \in T \mid Post(p,t) > 0\}$ ($P_i^\bullet = \{t \in T \mid Pre(p,t) > 0\}$). A *marked graph* (MG) is a P/T net such that each place has exactly one input arc and one output arc.

A *marking* is a vector $M : P \to I\!N$. $I\!N^{|P|}$ will denote the set of all possible markings that may be defined on the net. One writes $M\ [t\rangle\ M'$ to denote that an enabled t may fire at M yielding M'. $R(N, M_0)$ denotes the set of markings *reachable* on the net N from an initial marking M_0. A *P/T system* or *net system* $\langle N, M_0 \rangle$ is a net N with an initial marking M_0.

Let $\langle N, M_0 \rangle$ be a net system with set of places P. A single *generalized mutual exclusion constraint* $(\vec{w}, k)$ defines a set of legal markings: $\mathcal{M}(\vec{w}, k) = \{M \in I\!N^{|P|} \mid \vec{w}^T \cdot M \leq k\}$, where $\vec{w} : P \to I\!N$ is a weight vector, and $k \in I\!N^+$. The *support* of $\vec{w}$ is the set $Q_w = \{p \in P \mid w(p) > 0\}$. All markings that are not legal are called forbidden.

A set of GMEC $(W, \vec{k})$, with $W = [\vec{w}_1 \dots \vec{w}_m]$ and $\vec{k} = (k_1 \dots k_m)^T$, defines a set of legal markings $\mathcal{M}(W, \vec{k}) = \{M \in I\!N^{|P|} \mid W^T \cdot M \leq \vec{k}\}$.

As a particular case, when $\vec{w} \leq \vec{1}$, i.e., $w(p) = 1$ $(\forall p \in Q_w)$, the *unweighted* GMEC $(\vec{w}, k)$ is reduced to the *set condition* considered by Krogh and Holloway (1991).

We assume, now, that the set of transitions T of a net is partitioned into two disjoints subsets: T_u, the set of *uncontrollable* transitions; and T_c, the set of *controllable* transitions. A controllable transition may be disabled by the supervisor, a controlling agent which ensures that the behavior of the system is within a legal behavior. An uncontrollable transition represents an event which may not be prevented from occurring by a supervisor.

Given a system $\langle N, M_0 \rangle$ and a set of GMEC $(W, \vec{k})$, in the presence of uncontrollable transitions it is necessary to restrict the behavior of the system, avoiding not only all

forbidden markings but also the set $\mathcal{M}_u(W,\vec{k}) = \{M \in I\!N^{|P|} \mid M[\sigma\rangle M' \notin \mathcal{M}(W,\vec{k}) \wedge \sigma \in T_u^*\}$ of all those markings from which a forbidden marking may be reached by firing only uncontrollable transitions. The set of legal markings is in this case: $\mathcal{M}_c(W,\vec{k}) = \mathcal{M}(W,\vec{k}) \setminus \mathcal{M}_u(W,\vec{k})$.

3 Enforcing GMEC on Marked Graphs

3.1 Control Subnet

To enforce a GMEC it is necessary to prevent some transition firing. Although one cannot prevent the firing of an uncontrollable transition $t \in T_u$, one may prevent the firing of a set of controllable transitions (called *control transitions of* t) whose firing is required prior to the firing of t.

Definition 1. *Let $N = (P, T, Pre, Post)$ be a net, and let t_i be an uncontrollable transition. The* control subnet *for t_i is $N_i = (P_i, T_i, Pre_i, Post_i)$ where $P_i \subseteq P$ is the set of places connected to t_i by a path containing only uncontrollable transitions; $T_i = {}^\bullet P_i \cap P_i^\bullet$; $Pre_i = Pre \cap (P_i \times T_i)$, and $Post_i = Post \cap (P_i \times T_i)$. The set of* control transitions *for t_i is the set $A_i = {}^\bullet P_i \setminus P_i^\bullet$. It is obvious that $A_i \subseteq T_c$.*

This definition may be extended to controllable transitions as well. Given a net $N = (P, T, Pre, Post)$ and a controllable transition $t_i \in T_c$, the control subnet for t_i is not defined but the set of control transitions for t_i is the set $A_i = \{t_i\}$, i.e., the transition itself. This will allows us, in the following, to use the same formalism for both controllable and uncontrollable transitions.

In the case of marked graphs (MG), given a constraint $(\vec{w}, k)$ the problem is that of controlling the firing of the single input transition of all places in Q_w to ensure that the constraint is always verified. Given a place $p_i \in Q_w$ one may denote: t_i^o its output transition; t_i its input transition; $N_i = (P_i, T_i, Pre_i, Post_i)$ the control subnet for t_i; $A_i = \{t_i^1, \cdots, t_i^{n_i}\}$ the set of control transitions for t_i. Thus, one may speak of control subnets and control transitions associated to a place $p_i \in Q_w$.

Example 1. *In Fig. 1 places p_1, p_2, p_3 belong to the support of a GMEC. The figure also shows the control subnets for their input transitions, with the corresponding control transitions. The remaining structure of the marked graph is not shown. The controllable transitions are shown as white boxes; the uncontrollable ones are shown as black boxes.*

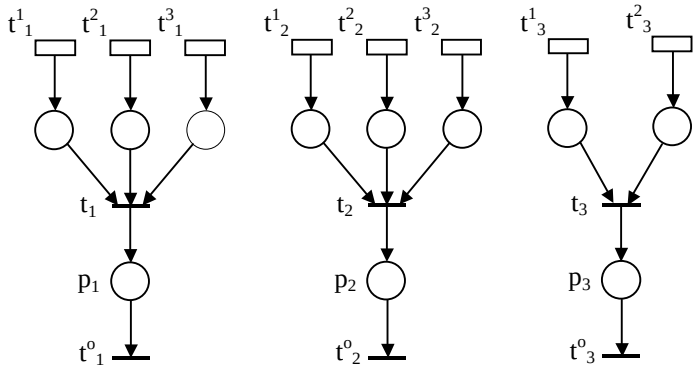

Figure 1: Control subnets for places p_1, p_2, p_3.

For MG systems it is possible to analytically compute the dependency between the firing of an uncontrollable transition and the firing of its control transitions. One may thus characterize the set of legal markings $\mathcal{M}_c(\vec{w}, k)$ for a GMEC $(\vec{w}, k)$ on marked graphs based solely on the structure of the net, without resorting to the construction of the space of reachable markings.

Here it is assumed that no place $p \in Q_w$ belongs to the control subnet of another place $p' \in Q_w$. In Giua (1992) it was shown that this restriction is purely technical and does not cause any loss of generality for the classes of nets considered in the following.

Proposition 1. *Let $\langle N, M_0 \rangle$ be an MG system and $(\vec{w}, k)$ a GMEC. For each place $p_i \in Q_w$, let its input transition be t_i; let A_i be the set of control transitions for t_i.*

- *Given a marking $M \in R(N, M_0)$, the maximum number of times t_i may fire without firing any transition in A_i (the* deviation bound *between t_i and A_i) is: $DB(M, t_i, A_i) = \min\{td(M, t', t_i) \mid t' \in A_i\}$, where $td(M, t', t_i)$ is the token distance between transitions t' and t_i, i.e., the minimum token content among all possible direct paths from t' to t_i at marking M.*
- *The set of legal markings is: $\mathcal{M}_c(\vec{w}, k) = \{M \in \mathbb{N}^{|P|} \mid \vec{w}^T \cdot (M + D_M) \leq k\}$, where $D_M(p_i) = DB(M, t^i, A^{t^i})$* **if** *$p_i \in Q_w$* **else** *$D_M(p_i) = 0$.*

3.2 Control Safe Places

Let us consider in the following a special class of MG systems introduced in the next definition.

Definition 2. *A place p_i of an MG system $\langle N, M_0\rangle$ is said to be* control safe *iff for all $M \in R(N, M_0)$ and for all $t' \in A_i$: $DB(M, t', t_i) \leq 1$ and $DB(M, t', t_i^o) \leq 1$ (here t_i and t_i^o are the input and ouptut transitions of p_i).*

Given a GMEC $(\vec{w}, k)$, it will be assumed in the following that the places in Q_w are control safe. This restriction will permit a simplification of the control problem, in the sense that will allow one to derive simple control structures. The idea here is that to check whether a place $p_i \in Q_w$ may be marked by a firing sequence of uncontrollable transitions one needs to check only how many firings of transitions in A_i have occurred.

Proposition 2. *Let $\langle N, M_0\rangle$ be a MG system, and p_i a control safe place of N. Let t_i be the input transition of p_i, t_i^o the output transition of p_i, and A_i the set of control transitions of t_i, with $n_i = |A_i|$. Then $\forall M \in R(N, M_0)$:*

- $M(p_i) + D_M(p_i) \leq 1$
- $M(p_i) + D_M(p_i) = 1 \Leftrightarrow (\forall t \in A_i)[td(M, t, t_i^o) = 1]$

4 Control Structures

This section presents two different ways of enforcing a GMEC on MG systems with control safe places. Both solutions are fully compiled, i.e., the corresponding control structure is a net system as well.

4.1 Monitor-based Controller

Definition 3. *Let $\langle N, M_0\rangle$ be a MG system and $(\vec{w}, k)$ be an unweighted mutual exclusion constraint, i.e., $\vec{w} \leq \vec{1}$, defined on it. It is assumed that $M_0 \in \mathcal{M}_c(\vec{w}, k)$. Let $Q_w = \{p_1, \ldots, p_r\}$ be a set of control safe places of N and assume that $r = |Q_w| = k + 1$. Given $p_i \in Q_w$ let $A_i = \{t_i^1, \ldots, t_i^{n_i}\}$ be the set of control transitions for t_i and t_i^o be the output transition of p_i. The* monitor *that enforces this constraint consists of a place S to be added to the original net with arcs as follows: $Pre(S, t) = 1$ **if** $t \in A_i$ **else** $Pre(S, t) = 0$; $Post(S, t) = n_i$ **if** $t = t_i^o$ **else** $Post(S, t) = 0$. The initial marking will assign $s - 1$ tokens to place S where $s = \mid \{t \mid t \in A_i \wedge td(M_0, t, t_i^o) = 0\} \mid$, i.e., s counts the number of control transitions that must fire prior to the marking of all places in Q_w.*

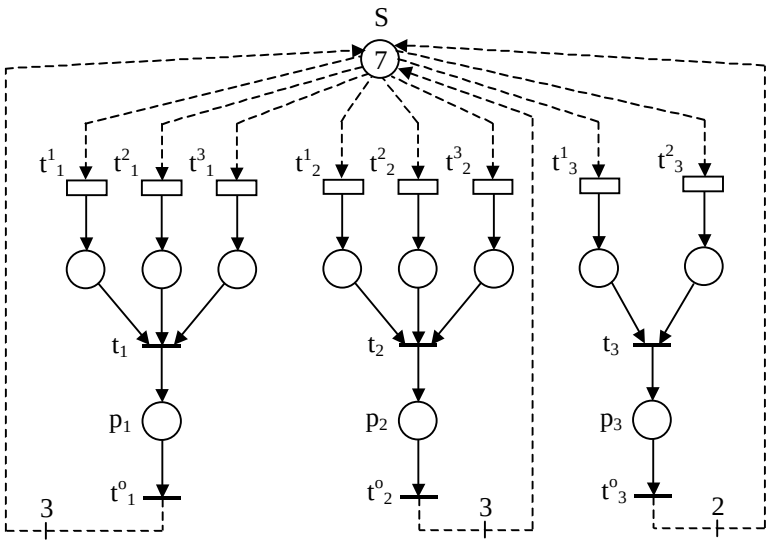

Figure 2: Example of monitor.

The previous definition assume that no transition will be selflooped with the monitor place. E.g., a transition t will be selflooped if $\exists i, j \ni t = t_i^o \wedge t \in A_j$. In this case one needs to eliminate the arcs in selfloop from the pre-incidence and post-incidence matrices.

Example 2. *Let us assume it is required to enforce the constraint* $M(p_1) + M(p_2) + M(p_3) \leq 2$ *over the net in Fig. 1. In Fig. 2 it is shown the monitor place* S *with its arcs shown in dotted lines.*

It is easy to see that a monitor constructed as in Definition 3 enforces the maximally permissible policy that ensures that the constraint $(\vec{w}, k)$ will be satisfied. In fact the monitor prevents only transition firings that lead to a marking M such that $(\forall i = 1, \ldots, r)(\forall t \in A_i)[td(M, t, t_i^o) = 1]$, that by Proposition 1 and 2 are the only illegal markings for unweighted constraints of this kind.

A set of constraints $(W, \vec{k})$ of this form may be enforced by adding several monitors.

Assume now $(\vec{w}, k)$, with $\vec{w} \leq \vec{1}$, is such that $|Q_w| > k + 1$. The previous construction may not be used. However the original constraint may be rewritten as a set of constraints according to the following proposition.

Proposition 3. Let $(\vec{w}, k)$ be a mutual exclusion constraint, with $\vec{w} \leq \vec{1}$, and $|Q_w| > k+1$. Then: $\mathcal{M}(\vec{w}, k) = \bigcap_{\vec{w}' \in I_{k+1}} \mathcal{M}(\vec{w}', k)$ where $I_{k+1} = \{\vec{w} \in \{0,1\}^{|P|} \mid \vec{w}' \leq \vec{w}, |Q_{w'}| = k+1\}$.

The previous proposition shows that the unweighted constraint $(\vec{w}, k)$, with $\vec{w} \leq \vec{1}$ and $|Q_w| > k+1$, is equivalent to the set of constraints $(W, k \cdot \vec{1}) = \{(\vec{w}', k) \mid \vec{w}' \in I_{k+1}\}$, hence

may be enforced by a set of monitors. However the problem is that there are $\begin{pmatrix} |Q_w| \\ k+1 \end{pmatrix}$ different subsets of Q_w of cardinality $k+1$. Thus in the worst case the number of monitors is exponential with respect to the cardinality of Q_w.

The monitor based construction may also be used, when the weight of the places is not unitary, explicitly rewriting the set of unweighted constraints equivalent to the single weighted constraint. Giua *et al.* (1992) proved that this is always possible for the class of nets considered here.

4.2 Compiled Supervisor

This section presents a net supervisor, capable of enforcing a set of GMEC. It is assumed that the supervisor observes the execution of the unconstrained system and at any given instant provides a control pattern, i.e., specifies which controllable transitions are allowed to fire. The control pattern is implicit in the transition structure of the supervisor, in the sense that a controllable transition that belongs to the supervisor structure is enabled by the control pattern if and only if it is enabled by the marking of the supervisor net.

Definition 4. *Let $\langle N, M_0 \rangle$ be a MG system and $(\vec{w}, k)$ be a GMEC. It is assumed that $M_0 \in \mathcal{M}_c(\vec{w}, k)$. Let $Q_w = \{p_1, \ldots, p_r\}$ be a set of control safe places. Given $p_i \in Q_w$ let $A_i = \{t_i^1, \ldots, t_i^{n_i}\}$ be the set of control transitions for t_i and t_i^o be the output transition of p_i. The* compiled supervisor *that enforces this constraint is $S = (P_S, T_S, Pre_S, Post_S)$ with: $P_S = \{p_0, p_1', p_1'', p_1''', p_2', \ldots, p_r', p_r'', p_r'''\}$; $T_S = A_1' \cup A_1'' \cup A_2' \ldots \cup A_r' \cup A_r'' \cup \{t_1^o, \ldots, t_r^o\}$ where A_i' and A_i'' are sets of transitions synchronized with A_i; Pre_S and $Post_S$ are such that:*

- $Pre(p_0, t) = w(p_i)$ **if** $t \in A_i''$ **else** $Pre(p_0, t) = 0$;
- $Post(p_0, t) = w(p_i)$ **if** $t = t_i^o$ **else** $Post(p_0, t) = 0$;
- $Pre(p_i', t) = 1$ **if** $t \in A_i'$ **else** $Pre(p_i', t) = 0$;
- $Post(p_i', t) = n_i - 1$ **if** $t = t_i^o$ **else** $Post(p_i', t) = 0$;
- $Pre(p_i'', t) = n_i - 1$ **if** $t \in A_i''$ **else** $Pre(p_i'', t) = 0$;
- $Post(p_i'', t) = 1$ **if** $t \in A_i'$ **else** $Post(p_i'', t) = 0$;

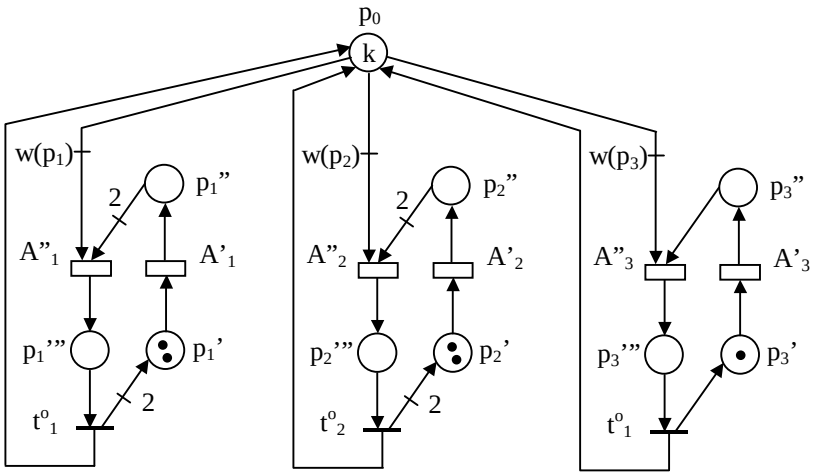

Figure 3: Example of compiled supervisor.

- $Pre(p_i''', t) = 1$ **if** $t = t_i^o$ **else** $Pre(p_i''', t) = 0$;
- $Post(p_i''', t) = 1$ **if** $t \in A_i''$ **else** $Post(p_i''', t) = 0$.

The initial marking of S *is* M_0^S *such that* $\forall i = 1, \ldots, r$: **if** $[M_0(p_i) + D_{M_0}(p_i) = 1]$ **then** $[M_0^S(p_i') = M_0^S(p_i'') = 0 \wedge M_0^S(p_i''') = 1]$ **else** $[M_0^S(p_i') =| \{t \in A_i \mid td(M_0, t, t_i^o) = 0\} \mid \wedge M_0^S(p_i'') = n_i - 1 - M_0^S(p_i') \wedge M_0^S(p_i''') = 0]$, *and* $M_0^S(p_0) = k - \sum_{i=1}^{r} w(p_i) M_0^S(p_i''')$.

Example 3. *In Fig. 3 it is shown the supervisor that may be used to enforce the constraint* $w(p_1)M(p_1) + w(p_2)M(p_2) + w(p_3)M(p_3) \leq k$ *over the net shown in Fig. 1.*

In the example in Fig. 3 the set of parallel transitions A_i' (and A_i'') have been represented as a single transition. Whenever the system executes a transition $t \in A_i$, the corresponding transition in A_i' or A_i'' will fire. Note that the behavior is deterministic: if a transition in A_i' is enabled, the corresponding transition in A_i'' is not, and conversely. For the computation of the control pattern, a transition in A_i is enabled by the control pattern if the corresponding transition in A_i' or in A_i'' is enabled.

In the previous definition there are two assumptions. Firstly, it is assumed that $(\forall i = 1, \ldots, r)$ $n_i > 1$; if $n_i = 1$ one may remove the places p_i' and p_i'' and the set of transitions A_i'. Secondly, it is assumed that $(\forall i \neq j)$ $A_i \cap A_j = \emptyset$; if this is not the case, we need to slightly change the structure of the supervisor by merging the transitions of A_i' and A_i'' in common with A_j' and A_j''.

The supervisor constructed according to Definition 4 enforces the required maximally permissible policy. First, let us note that given a marking M of the system and a correspond-

ing marking M^S of the supervisor it holds: $(\forall i = 1, \ldots, r)[M(p_i) + D_M(p_i) = M^S(p_i''')]$. Since the place p_0 is enforcing the constraint $\sum_{i=1}^{r} w(p_i) M^S(p_i''') \leq k$ it follows, by Proposition 1 and 2, that the supervisor enforces the required policy.

In the case of a set of constraints $(W, \vec{k})$ one needs to construct a supervisor for each single constraint $(\vec{w}_i, k_i)$. Should a controllable transition belong to more that one supervisor, say S_{i_1} and S_{i_2}, it will be enabled by the control pattern if and only if it is enabled on both S_{i_1} and S_{i_2}.

4.3 Comparison of the Models

The *monitor-based controller* is an extension to systems with uncontrollable transitions of the controller studied by Giua *et al.* (1992) for nets with only controllable transitions. Thus all the structural properties of monitors may be used to analyze the system under control. The drawback is that a monitor-based solution may require an exceedingly large number of monitors. However, in those cases in which it may be used efficiently, it is the simpler and most straightforward solution.

The *compiled supervisor* has the advantage of always maintaining a compact structure that grows linearly with the number of places in the support of the weight vector. However, since it requires all control transitions to be represented twice, it leads to a closed-loop model less easy to analyze.

5 Conclusion

The paper has presented two Petri net structures capable of enforcing generalized mutual exclusion constraints on marked graphs with control safe places and uncontrollable transitions. The first one, is a monitor-based solution; the second one a supervisory based. Both structures are fully compiled, i.e., they are given as place/transition nets with no associated predicates.

The monitor-based structure is conceptually simpler but in the worst case it is requires a number of monitors that grows exponentially with the number of places in the support of the constraint weight vector. The supervisory based structure grows linearly with the number of places in the support of the weight vector, thus always maintaining a compact

control structure.

References

Giua, A. (1992). Petri Nets as Discrete Event Models for Supervisory Control. *Doctoral Thesis*, ECSE Dept., Rensselaer Polytechnic Institute, Troy, New York.

Giua, A., F. DiCesare, and M. Silva (1992). Generalized Mutual Exclusion Constraints on Nets with Uncontrollable Transitions. *Proc. 1992 IEEE Int. Conf. on Systems, Man, and Cybernetics* (Chicago, Illinois).

Golaszewski, C.H., and P.J. Ramadge (1988). Mutual Exclusion Problems for Discrete Event Systems with Shared Events. *Proc. IEEE 27th Int. Conf. on Decision and Control* (Austin, Texas), 234–239.

Krogh, B.H., and L.E. Holloway (1991). Synthesis of Feedback Control Logic for Discrete Manufacturing Systems. *Automatica*, Vol. 27, No. 4, 641–651.

Murata, T. (1989). Petri Nets: Properties, Analysis and Applications. *Proceedings IEEE*, Vol. 77, No. 4, 541–580.

Ramadge, P.J., and W.M. Wonham (1989). The Control of Discrete Event Systems. *Proceedings IEEE*, Vol. 77, No. 1, 81–98.