Suboptimal Supervisory Control of Petri Nets in presence of Uncontrollable Transitions via Monitor Places

Francesco Basile [1] , Pasquale Chiacchio

Dip. di Ing. dell'Informazione ed Ing. Elettrica, Università di Salerno, Via Ponte Don Melillo, 84084 Fisciano (SA), Italy

Alessandro Giua

Dip. di Ing. Elettrica ed Elettronica, Università di Cagliari, Piazza d'Armi, 09123 Cagliari, Italy

Abstract

This paper deals with the problem of enforcing generalized mutual exclusion constraints (GMEC) on place/transition nets with uncontrollable transitions. An efficient control synthesis technique, that has been proposed in the literature, is to enforce GMEC constraints by introducing monitor places to create suitable place invariants. The method has been shown to be maximally permissive and to give a unique control structure in the case that the set of legal markings is controllable. This paper investigates on and formally shows that the class of controllers obtained by this technique may not have a supremal element for uncontrollable specifications. Moreover, it is shown that the family of monitor places enforcing an uncontrollable specification can be parameterized with respect to the solution of a linear system of equation. An algorithm to obtain such parameterization is here presented.

Key words: discrete event systems, supervisory control, petri nets, monitors

1 Introduction

In the original approach of Ramadge and Wonham (1989) to the supervisory control of discrete event systems (DESs), a DES G is a language generator whose behaviour, i.e., language, is denoted $L(G)$. Given a legal language L, the

[1] Corresponding author, e-mail: fbasile@unisa.it

Accepted for publication as Brief paper to Automatica *November 2005*

basic control problem is to design a supervisor that restricts the closed loop behaviour of the plant to $L \cap L(G)$, disabling or enabling controllable events; the events whose occurrence cannot be disabled are called uncontrollable. This is possible if and only if two conditions are met: L must be prefix-closed and controllable. The first condition is purely technical and will not be discussed any further. The language controllability property is, on the contrary, much more interesting. If L is not controllable, we can consider the class of prefix closed and controllable sublanguages of L, i.e., the set $\Omega(L) = \{K \subseteq L \mid K \text{ is prefix closed and controllable}\}$. For each language K in this class we may construct a supervisor, thus further restricting the closed loop behaviour of the plant to $K \cap L(G) \subset L \cap L(G)$. The class $\Omega(L)$ is closed under union and not empty, hence it admits a unique maximal (i.e. supremal) element with respect to set inclusion. The element $L^{\uparrow} = \sup \Omega(L)$, called *supremal controllable sublanguage*, is the "optimal" solution to our control problem in the sense that it is the minimally restrictive solution.

A similar approach can also be taken when considering the state evolution of a DES, rather than the traces of events it generates. This approach, that we call *state-based*, is particularly attractive when Petri nets (PNs) are used to represent the plant (Krogh and Holloway, 1991; Li and Wonham, 1994; Zhou and DiCesare, 1993). In this case it is assumed that some transitions, that we call controllable, can be disabled by an external agent. Let us consider a PN system $\langle N, \boldsymbol{m}_0 \rangle$ with m places, whose set of reachable markings is $R(N, \boldsymbol{m}_0) \subseteq \mathbb{N}^m$. Assume we are given a set of legal markings $\mathcal{L} \subseteq \mathbb{N}^m$, and consider the basic control problem of designing a supervisor that restricts the reachability set of plant in closed loop to $\mathcal{L} \cap R(N, \boldsymbol{m}_0)$. This is possible if and only if two conditions are met: $\mathcal{L}$ must be legally reachable and controllable. The first condition requires that all legal markings may be reached through a state evolution that does not contain forbidden markings: this condition, in a similar way to the prefix-closure property mentioned above, is purely technical and will not be discussed any further. The state controllability property is, on the contrary, much more interesting and will be formally given in Definition 2. If $\mathcal{L}$ is not controllable, we can consider the class of controllable subsets of $\mathcal{L}$, i.e., the class $\Omega(\mathcal{L}) = \{\mathcal{K} \subseteq \mathcal{L} \mid \mathcal{K} \text{ is controllable}\}$. For each set $\mathcal{K}$ in $\Omega(\mathcal{L})$ we may construct a supervisor, thus further restricting the reachability set of the plant in closed loop to $\mathcal{K} \cap R(N, \boldsymbol{m}_0) \subset \mathcal{L} \cap R(N, \boldsymbol{m}_0)$. The class $\Omega(\mathcal{L})$ is closed under union and not empty [2] hence it admits a unique supremal element with respect to set inclusion. The element $\mathcal{L}^{\uparrow} = \sup \Omega(\mathcal{L})$, called *supremal controllable subset*, is the "optimal" solution to this control problem in the sense that it allows the largest closed loop reachability set.

[2] This is true under the non-concurrency hypothesis. In the approach of Holloway and Krogh two transitions may fire concurrently and this is not true anymore (Holloway *et al.*, 1997).

Of particular interest are those PN state-based control problems where the set of legal markings $\mathcal{L}$ is expressed by a set of n_c linear inequality constraints called Generalized Mutual Exclusion Constraint (GMEC). In this case we write $\mathcal{L} = \mathcal{M}(\boldsymbol{L}, \boldsymbol{k}) \equiv \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{Lm} \leq \boldsymbol{k}\}$ to denote that $\mathcal{L}$ is expressed by the GMEC $(\boldsymbol{L}, \boldsymbol{k})$ with $\boldsymbol{L} \in \mathbb{Z}^{n_c \times m}, \boldsymbol{k} \in \mathbb{Z}^{n_c}$. Problems of this kind have been considered by several authors (Giua *et al.*, 1992; Moody *et al.*, 1996; Li and Wonham, 1994; Krogh and Holloway, 1991). This special structure of the legal set has the advantage that if $\mathcal{L}$ is controllable then the supervisor for this class of problems takes the form of as many places, called *monitors*, as there are constraints. Thus if the matrix $\boldsymbol{L}$ has n_c rows, the supervisor will consist of n_c monitor places, each of which has arcs going to and coming from some transitions of the plant net. The DES plant and the controller are described by PNs in order to have a useful linear algebraic model for control analysis and synthesis. Moreover the synthesis is not computationally demanding since it involves only a matrix multiplication.

Let us assume, however, that $\mathcal{L}$ is uncontrollable. Following the general approach outlined above, we have to compute the set $\mathcal{L}^{\uparrow}$. The supervisory synthesis becomes now complex because we have to deal with controllable subsets. Furthermore, in most cases the special structure of the legal set is lost, because, as shown by Giua *et al.* (1992), it may well be the case that $\mathcal{L}^{\uparrow}$ cannot be expressed by a set of linear inequalities, i.e., the corresponding supervisor does not have a monitor-based structure. Only in a few cases this special structure is kept. In Li and Wonham (1994) it was shown that if the plant net belongs to the special class of TS2 nets then $\mathcal{L}^{\uparrow}$ is guaranteed to be expressed by a set of n_c linear inequalities. In Giua *et al.* (1992) it was shown that if the plant net is safe then $\mathcal{L}^{\uparrow}$ is guaranteed to be expressed by a set of n_c' linear inequalities, where n_c', however, may be very large (it may be of the same order of the cardinality of the reachability set).

This problem motivated Moody et al. (1996; 2000) to consider as acceptable a further restriction of the reachability set. The idea is that of finding a subset of $\mathcal{L}^{\uparrow}$ such that: (a) it can be guaranteed to be controllable by structural conditions; (b) it can be expressed by a set of n_c constraints. In particular the structural controllability condition requires that no monitor place has arcs going to an uncontrollable transition so that it may never prevent its firing. Thus given an uncontrollable legal marking set $\mathcal{L}$ expressed by n_c constraints, one may define the set $\Omega_{n_c}(\mathcal{L}) = \{\mathcal{K} \subseteq \mathcal{L} \mid \mathcal{K} \text{ is structurally controllable}, \exists \boldsymbol{L}' \in \mathbb{Z}^{n_c \times m}, \boldsymbol{k}' \in \mathbb{Z}^{n_c} : \mathcal{K} = \mathcal{M}(\boldsymbol{L}', \boldsymbol{k}')\}$ of *structurally controllable* and *expressed by a set of* n_c *linear inequalities* subsets of $\mathcal{L}$. In (Moody *et al.*, 1996; Moody and Antsaklis, 2000) a procedure was also given that leads to compute an element $\mathcal{K} \in \Omega_{n_c}(\mathcal{L})$, i.e., to compute a constraint $(\boldsymbol{L}', \boldsymbol{k}')$ with $\boldsymbol{L}' \in \mathbb{Z}^{n_c \times m}$, and its corresponding monitor structure, such that $\mathcal{K} = \mathcal{M}(\boldsymbol{L}', \boldsymbol{k}')$. We note that in this approach one restricts the reachability set of the plant in closed loop to be within $\mathcal{K} \subset \mathcal{L}^{\uparrow}$, i.e., one may prevent the closed loop system from reaching some perfectly legal markings. One gains, however, in simplicity since the controller takes a simple structure of n_c monitors and the controllability condition can be ensured without resorting to study the reachability set of the net. A different technique to compute a set of controllable constraints has been also presented in (Stremersch, 2000). In this work the transformed constraints were not obtained by working on the net structure and the controllable constraints were found by using the relaxation that the constraint weights are rational, hence there was not possible to characterize maximal elements.

In this paper we further pursue the investigation along these lines and we present the following results:

- A formal proof that the class $\Omega_{n_c}(\mathcal{L})$ is not closed under union and not empty. Hence a maximal element exists but it is not necessarily unique.
- An algorithm to construct a parameterization of monitors corresponding to a set of constraints that includes the *maximal* elements of $\Omega_{n_c}(\mathcal{L})$ (and thus they are structurally controllable), when $\mathcal{L} = \mathcal{M}(\boldsymbol{L}, \boldsymbol{k})$ with $\boldsymbol{L} \in \mathbb{N}^{n_c \times m}, \boldsymbol{k} \in \mathbb{N}^{n_c}$. This parameterization takes the form of a unique control net incidence matrix that depends linearly on the value of the parameters subject to a linear equations system.

2 Background

A place/transition (P/T) net is a structure $N = \langle P, T, \mathbf{Pre}, \mathbf{Post} \rangle$ where: P is a set of m *places* represented by circles; T is a set of n *transitions* represented by bars; $P \cap T = \emptyset$, $P \cup T \neq \emptyset$; $\mathbf{Pre}$ ($\mathbf{Post}$) is the $\mid P \mid \times \mid T \mid$ sized, natural valued, pre-(post-)incidence matrix. The incidence matrix $\boldsymbol{C}$ of the net is defined as $\boldsymbol{C} = \mathbf{Post} - \mathbf{Pre}$. For pre- and post-sets we use the dot notation, e.g. ${}^\bullet t = \{p \in P \mid \mathbf{Pre}(p, t) \neq 0\}$. A *marking* is a $m \times 1$ vector $\boldsymbol{m} : P \to \mathbb{N}$ that assigns to each place of a P/T net a non-negative integer number of tokens. A P/T system $\langle N, \boldsymbol{m}_0 \rangle$ is a P/T net N with an initial marking $\boldsymbol{m}_0$. A marking $\boldsymbol{m}$ is reachable in $\langle N, \boldsymbol{m}_0 \rangle$ iff there exists a firing sequence $\sigma = t_1 ... t_k$ such that $\boldsymbol{m}_0[\sigma > \boldsymbol{m}$, the set of reachable markings in $\langle N, \boldsymbol{m}_0 \rangle$ is denoted $R(N, \boldsymbol{m}_0)$. Consider a set of legal markings $\mathcal{L} \subseteq \mathbb{N}^m$ and the control problem of designing a supervisor that restricts the plant reachability set in closed loop to $\mathcal{L} \cap R(N, \boldsymbol{m}_0)$. Of particular interest are those PN state-based control problems where the legal marking set $\mathcal{L}$ is expressed by a set of n_c GMECs. Given the net system $\langle N, \boldsymbol{m}_0 \rangle$, a GMEC is a couple $(\boldsymbol{l}, k)$ where $\boldsymbol{l} : P \to \mathbb{Z}$ is a $1 \times m$ weight vector and $k \in \mathbb{Z}$. The support of $\boldsymbol{l}$ is the set $Q_l = \{p \in P \mid \boldsymbol{l}(p) \neq 0\}$. The set of *legal markings* defined by $(\boldsymbol{l}, k)$ is $\mathcal{M}(\boldsymbol{l}, k) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{l}\boldsymbol{m} \leq k\}$. A set of GMEC $(\boldsymbol{L}, \boldsymbol{k})$, with $\boldsymbol{L} = [\boldsymbol{l}_1^T, \boldsymbol{l}_2^T, ... \boldsymbol{l}_{n_c}^T]^T$ and $\boldsymbol{k} = [k_1, k_2, ... k_{n_c}]^T$, will define the *legal markings set* $\mathcal{M}(\boldsymbol{L}, \boldsymbol{k}) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{L}\boldsymbol{m} \leq \boldsymbol{k}\}$.

The markings that are not legal are called *forbidden markings*. A controlling agent, called supervisor, must ensure that the forbidden markings will not be reached, then the set of legal markings under control is $\mathcal{M}_c(\boldsymbol{L}, \boldsymbol{k}) = \mathcal{M}(\boldsymbol{L}, \boldsymbol{k}) \cap R(N, \boldsymbol{m}_0)$. If all transitions are controllable, Giua *et al.* (1992) showed that the PN controller enforcing $(\boldsymbol{L}, k)$ has incidence matrix $\boldsymbol{C}_c \in \mathbb{Z}^{n_c \times n}$ given by $\boldsymbol{C}_c = -\boldsymbol{L}\boldsymbol{C}_p$ where $\boldsymbol{C}_p$ is the incidence matrix of the plant and the initial marking of the controller $\boldsymbol{m}_{c0} \in \mathbb{N}^{n_c \times 1}$ is given by $\boldsymbol{m}_{c0} = \boldsymbol{k} - \boldsymbol{L}\boldsymbol{m}_{p0}$ where $\boldsymbol{m}_{p0} \in \mathbb{N}^{m \times 1}$ is the initial marking of the plant. The controller exists iff the initial marking is legal, i.e. $\boldsymbol{k} - \boldsymbol{L}\boldsymbol{m}_{p0} \geq \boldsymbol{0}$. The controller so constructed is maximally permissive, i.e. it prevents only transitions firings that yield forbidden markings. The control net has n_c control places, called *monitor places*; no transition is added. The i-th monitor place, denoted as p_{ci}, is connected to transition plant t_j as specified by an arc of weight $\boldsymbol{C}_c(p_{ci}, t_j) = \boldsymbol{L}(i, \cdot)\boldsymbol{C}_p(\cdot, t_j)$. Consider the net system in fig. 1a. Assume that all the transitions are controllable. Let $\mathcal{L} = \mathcal{M}(\boldsymbol{l}, k) = \{\boldsymbol{m} \in \mathbb{N}^m \mid m(p_3) + m(p_4) \leq 1\}$. Thus, the monitor p_c in fig. 1b is derived since $\boldsymbol{C}_c = [\,-1 \quad -1 \quad 1 \quad 1\,]$ and $\boldsymbol{m}_0(p_c) = [\,1\,]$.

3 Monitor synthesis in presence of uncontrollable transitions

Let us now consider the problem of restricting the reachability set of a PN within a set of legal markings $\mathcal{L}$ in presence of uncontrollable transitions. The set of transitions T is partitioned in two disjoint subsets: T_u, the set of uncontrollable transitions, associated to uncontrollable events and drawn as black bars, and T_c, the set of controllable transitions, associated to controllable events and drawn as empty boxes.

Definition 1 *Consider a net N with $T_c \neq \emptyset$ and a GMEC $(\boldsymbol{l}, k)$. We define the uncontrollable subnet of N, denoted as $N_u = \langle P, T_u, \mathbf{Pre}_u, \mathbf{Post}_u \rangle$, the subnet obtained from N eliminating every controllable transition.*

It is immediate to see that $R(N_u, \boldsymbol{m}) \subseteq R(N, \boldsymbol{m})$. The uncontrollable subnet definition let us to define when a set of legal markings $\mathcal{L} \subseteq \mathbb{N}^m$ is controllable.

Definition 2 *A legal marking set $\mathcal{L} \subseteq \mathbb{N}^m$ is controllable w.r.t. a PN system $\langle N, \boldsymbol{m}_0 \rangle$ with uncontrollable subnet N_u if $\bigcup_{\boldsymbol{m} \in \mathcal{L} \cap R(N, \boldsymbol{m}_0)} R(N_u, \boldsymbol{m}) \subseteq \mathcal{L}$.*

According to this definition, $\mathcal{L}$ is controllable if from any marking $\boldsymbol{m} \in \mathcal{L}$ no forbidden marking is reachable by firing a sequence containing only uncontrollable transitions, that cannot be disabled by a supervisor. Note that checking this condition requires, in general, computing the reachability set of the controlled net; only if N_u has a special structure, this computation can be done with structural analysis.

When the controller is modeled by a PN structure, the disabling of transition t is possible only if there is an arc from a controller place to t and the marking of the controller place does not enable the transition. If GMECs are considered, since a monitor place is needed for each GMEC, a restriction can be considered: arcs directed from monitor places to uncontrollable transitions have to be avoided in order to prevent that a monitor may disable an uncontrollable transition. It is very efficient from a computational point of view to impose this condition, as shown in Moody and Antsaklis (2000), that we call *structural controllability* condition, since the arcs directed from monitors to transition can be obtained by a simple matrix multiplication.

Definition 3 *Given a set of legal marking represented by a GMEC* $\mathcal{L} = \mathcal{M}(\boldsymbol{L}, k)$ *and a P/T net* N *having a transition set* $T = T_c \cup T_u$ *with* $T_c \cap T_u = \emptyset$, $\mathcal{L}$ *is structurally controllable if* $\boldsymbol{L}\boldsymbol{C}_u \leq \boldsymbol{0}$ *where* $\boldsymbol{C}_u \in \mathbb{Z}^{m \times n_u}$ *is the incidence matrix of the uncontrollable subnet* N_u *and* n_u *is the number of uncontrollable transitions of the plant net.*

The structural controllability represents a sufficient condition for the behavioral controllability (Ghaffari *et al.*, 2003). The monitor place p_c in fig. 1b) enforces $\boldsymbol{m}(p_3) + \boldsymbol{m}(p_4) \leq 1$ on the plant net system in fig. 1a). It does not meet the structural controllability condition because of of the arc from p_c to t_2. In fig. 1c) and d) the closed-loop reachability graph for $x = 1$ and $x = 2$ respectively are shown. Notice that the monitor disable the uncontrollable transition only if for $x = 2$. Thus the specification is controllable in the behavioral sense. In this paper we consider structural controllability.

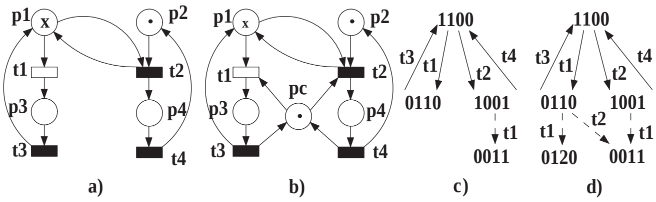


Fig. 1. a) a net system; b) $\boldsymbol{m}(p_3) + \boldsymbol{m}(p_4) \leq 1$ has been forced on net system in a) by monitor p_c; c) reachability graph of the controlled system in the case $x = 1$ (the transition firings disabled by the monitor are shown as dashed arcs); d) reachability graph of the controlled system in the case $x = 2$.

If $\mathcal{L}$ is not controllable, we also must avoid reaching the set of markings $\mathcal{L}_{uf} = \{\boldsymbol{m} \in \mathcal{L} \mid \boldsymbol{m}[\sigma > \boldsymbol{m}',\ \boldsymbol{m}' \notin \mathcal{L}, \sigma \in T_u^*\}$. We can consider the class of controllable subsets of $\mathcal{L}$, i.e., the class $\Omega(\mathcal{L}) = \{\mathcal{K} \subseteq \mathcal{L} \mid \mathcal{K}$ is controllable$\}$. The element $\mathcal{L}^{\uparrow} = \sup \Omega(\mathcal{L}) = \mathcal{L} \setminus \mathcal{L}_{uf}$, called *supremal controllable subset*, is the "optimal" solution to the control problem of restricting the reachability set of plant to legal markings. In the case of legal sets given by GMEC, Moody *et al.* (1996) proposed an efficient way to compute an approximation of $\mathcal{L}^{\uparrow}$, transforming the control specification GMEC $(\boldsymbol{L}, \boldsymbol{k})$ into a more restrictive GMEC $(\boldsymbol{L}', \boldsymbol{k}')$ as shown in the next proposition.

Proposition 4 (Moody *et al.* (1996)) *If we are able to find* $\boldsymbol{R}_1 \in \mathbb{N}^{n_c \times m}$, $\boldsymbol{R}_2 \in \mathbb{N}^{n_c \times n_c}$ *satisfying* $[\,\boldsymbol{R}_1 \quad \boldsymbol{R}_2\,] \begin{bmatrix} \boldsymbol{m}_{p0} \\ \boldsymbol{L}\boldsymbol{m}_{p0} - (\boldsymbol{k}+\boldsymbol{1}) \end{bmatrix} \leq -\boldsymbol{1}$ *then the controller computed as* $\boldsymbol{C}_c = -\boldsymbol{L}'\boldsymbol{C}_p$, $\boldsymbol{m}_{c0} = \boldsymbol{k}' - \boldsymbol{L}'\boldsymbol{m}_{p0}$ *where* $\boldsymbol{L}' = \boldsymbol{R}_1 + \boldsymbol{R}_2\boldsymbol{L}$, $\boldsymbol{k}' = \boldsymbol{R}_2(\boldsymbol{k}+\boldsymbol{1}) - \boldsymbol{1}$ *will be able to ensure that the closed loop net system meet* $\boldsymbol{L}\boldsymbol{m}_p \leq \boldsymbol{k}$*, and that the initial marking is legal.*

4 A first result: there is not a supremal monitor for uncontrollable specifications

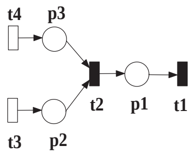


Fig. 2. A P/T net with uncontrollable transition t_2.

From now on we consider legal sets given by GMEC, i.e., $\mathcal{L}$ is expressed by a set of n_c linear inequality constraints and can be written as $\mathcal{L} = \mathcal{M}(\boldsymbol{L}, \boldsymbol{k}) \equiv \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{L}\boldsymbol{m} \leq \boldsymbol{k}\}$. If $\mathcal{L}$ is not controllable, as discussed in the introduction, $\mathcal{L}^{\uparrow}$ may not be expressed by a set of n_c linear inequality constraints. In this case, one may define the set $\Omega_{n_c}(\mathcal{L}) = \{\mathcal{K} \subseteq \mathcal{L} \mid \exists \boldsymbol{L}' \in \mathbb{Z}^{n_c \times m}, \boldsymbol{k}' \in \mathbb{Z}^{n_c} : \mathcal{K} = \mathcal{M}(\boldsymbol{L}', \boldsymbol{k}'), \mathcal{K} \text{ is structurally controllable}\}$ of *structurally controllable* and *expressed by a set of* n_c *linear inequalities* subsets of $\mathcal{L}$.

Theorem 5 *Consider a plant represented by a PN system* $\langle N, \boldsymbol{m}_0 \rangle$*. Let* $\mathcal{L} = \mathcal{M}(\boldsymbol{L}, \boldsymbol{k}) \equiv \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{L}\boldsymbol{m} \leq k\}$ *be an uncontrollable set with* $\boldsymbol{L} \in \mathbb{Z}^{n_c \times m}$ *and* $\boldsymbol{k} \in \mathbb{Z}^{n_c}$*. The class* $\Omega_{n_c}(\mathcal{L})$ *is:*
a) not empty;
b) not closed under union.

PROOF. a) Let us consider the set $\mathcal{K} = \emptyset \subset \mathcal{L}$. By definition 2, $\mathcal{K}$ is controllable. It can also be expressed by a set of linear inequalities: take any constraint set with no feasible solution. E.g., if we let $\boldsymbol{L}' = \{0\}^{n_c \times m}$ and $\boldsymbol{k}' = \{-1\}^{n_c}$, clearly $\mathcal{K} = \mathcal{M}(\boldsymbol{L}', \boldsymbol{k}')$. This shows that $\emptyset \in \Omega_{n_c}(\mathcal{L})$.

b) We show this giving a simple counterexample. Consider the net in fig. 2 with $T_u = \{t_2\}$. Let $\mathcal{L} = \{\boldsymbol{m} \in \mathbb{N}^3 \mid \boldsymbol{m}(p_1) \leq 1\}$. This set is not structurally controllable, because the corresponding monitor requires an arc going to the uncontrollable transition t_2.
Consider the sets: $\mathcal{K}_1 = \{\boldsymbol{m} \in \mathbb{N}^3 \mid \boldsymbol{m}(p_1) + \boldsymbol{m}(p_2) \leq 1\}$ and $\mathcal{K}_2 = \{\boldsymbol{m} \in \mathbb{N}^3 \mid \boldsymbol{m}(p_1) + \boldsymbol{m}(p_3) \leq 1\}$. Clearly, $\mathcal{K}_1, \mathcal{K}_2 \in \Omega_{n_c}(\mathcal{L})$.

We will show that the set $\mathcal{K} = \mathcal{K}_1 \cup \mathcal{K}_2$ is not convex, hence it cannot be expressed by a set of linear inequalities. In fact, if we consider the markings $\boldsymbol{m}_1 = [1\ 0\ 2]^T \in \mathcal{K}_1 \subset \mathcal{K}$ and $\boldsymbol{m}_2 = [1\ 2\ 0]^T \in \mathcal{K}_2 \subset \mathcal{K}$, the marking $\boldsymbol{m} = \frac{\boldsymbol{m}_1 + \boldsymbol{m}_2}{2} = [1\ 1\ 1]^T$ does not belong to $\mathcal{K}$. $\square$

Note that the part a) of the previous theorem shows that $\Omega_{n_c}(\mathcal{L})$ is not empty because it contains the empty set. However, if the supremal element of $\Omega_{n_c}(\mathcal{L})$ is the set $\mathcal{K} = \emptyset$, the (monitor-based) control problem has no solution, because the required condition that $\boldsymbol{m}_0 \in \mathcal{K}$ is clearly not satisfied.

Corollary 6 *Let consider a PN system* $\langle N, \boldsymbol{m}_0 \rangle$. *Let* $\mathcal{L} = \mathcal{M}(\boldsymbol{L}, \boldsymbol{k}) \equiv \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{L}\boldsymbol{m} \leq k\}$ *be an uncontrollable set with* $\boldsymbol{L} \in \mathbb{Z}^{n_c \times m}$ *and* $\boldsymbol{k} \in \mathbb{Z}^{n_c}$. *A maximal element of the set* $\Omega_{n_c}(L)$ exists but it is not necessary unique*.*

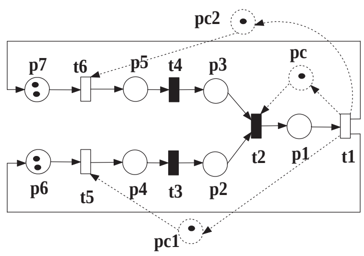


Fig. 3. Net system

5 A second result: a linear parameterization for the maximal monitor based controllers family

Consider the net system in fig. 3 without dashed arcs and places. Let $\mathcal{L} = \mathcal{M}(\boldsymbol{l}, k) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{m}(p_1) \leq 1\}$. The monitor p_c is derived to enforce $\mathcal{M}(\boldsymbol{l}, k)$. It does not meet the structural controllability condition because of the arc from p_c to t_2. A controllable subset of the uncontrollable specification $\mathcal{L} = \mathcal{M}(\boldsymbol{l}, k) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{m}(p_1) \leq 1\}$ can be expressed in terms of GMEC $\mathcal{M}(\boldsymbol{l}^1, k^1) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{m}(p_1) + \boldsymbol{m}(p_2) + \boldsymbol{m}(p_4) \leq 1\}$ where $\boldsymbol{l}^1 = r_2^1 \boldsymbol{l} + \boldsymbol{r}_1^1$ with $\boldsymbol{r}_1^1 = [0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0]$ and $r_2^1 = 1$. The output arc from p_c to the uncontrollable transition t_2 has been moved up into an output arc from p_{c1} to the controllable transition t_5. In this way the firing of t_2 (and thus the marking of p_1) is controlled by the marking of places that belong to a controlled path ending with t_2, and thus they are included in the support of the transformed constraint. Notice that the controlled path $t_6 p_5 t_4 p_3 t_2$ too can be used to control the firing of t_2. This can be done by enforcing another controllable subset of $\mathcal{M}(\boldsymbol{l}, k)$ that can be expressed in terms of GMEC too as $\mathcal{M}(\boldsymbol{l}^2, k^2) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{m}(p_1) + \boldsymbol{m}(p_3) + \boldsymbol{m}(p_5) \leq 1\}$ where $\boldsymbol{l}^2 = r_2^2 \boldsymbol{l} + \boldsymbol{r}_1^2$ with $\boldsymbol{r}_1^2 = [0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0]$ and $r_2^2 = 1$; the monitor p_{c2} can be derived to enforce $\mathcal{M}(\boldsymbol{l}^2, k^2)$. It is immediate to see that $\mathcal{M}(\boldsymbol{l}^2, k^2)$ and $\mathcal{M}(\boldsymbol{l}^1, k^1)$ are incomparable, i.e. $\mathcal{M}(\boldsymbol{l}^2, k^2) \nsubseteq \mathcal{M}(\boldsymbol{l}^1, k^1)$ and $\mathcal{M}(\boldsymbol{l}^2, k^2) \nsupseteq \mathcal{M}(\boldsymbol{l}^1, k^1)$. We can choose either one of the two monitors if we aim only to enforce $(\boldsymbol{l}, k)$ on the closed loop net system. However, if an additional control specification is present, e.g. liveness constraint or the cost of disabling controllable transitions etc., one of the two monitors may be preferable to the other one. Thus, it can be useful to represent in a compact form the constraint transformations to which correspond the monitors family that meet structural controllability. By introducing two parameters α_1, α_2, we can obtain the following compact form:

$$\begin{aligned}
&\boldsymbol{l}' = \boldsymbol{r}_1 + r_2\boldsymbol{l}, \quad \boldsymbol{k}' = r_2(k+1) - 1\\
&with \quad \boldsymbol{r}_1 = [0 \quad \alpha_1 \quad \alpha_2 \quad \alpha_1 \quad \alpha_2 \quad 0 \quad 0], \quad r_2 = 1\\
&s.t. \begin{cases} (a) \quad \alpha_1 + \alpha_2 = 1 \\ (b) \quad \alpha_i \in \mathbb{N} \end{cases}
\end{aligned} \tag{1}$$

The condition (1-a) represents the choice to control t_2 by t_5 or t_6 that both control the token flow in paths terminating with t_2. The dependence from the α parameters above can be presented in the following linear form

$$\begin{aligned}
&\boldsymbol{l}' = \boldsymbol{r}_1 + r_2\boldsymbol{l}, \quad \boldsymbol{k}' = r_2(k+1) - 1\\
&with \quad \boldsymbol{r}_1 = \boldsymbol{q} + \boldsymbol{\alpha}^T\boldsymbol{P}, \quad r_2 \in \mathbb{N}\\
&s.t. \begin{cases} \boldsymbol{A}\boldsymbol{\alpha} = \boldsymbol{b} \\ \boldsymbol{\alpha} \in \mathbb{N}^2 \end{cases}
\end{aligned} \tag{2}$$

where $\boldsymbol{q} = \boldsymbol{0}_{1\times 7}$, $\boldsymbol{P} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$, $\boldsymbol{A} = [1 \quad 1]$, $\boldsymbol{b} = [1]$. In the rest of the section this idea is formalized and an algorithm, that finds all possible constraint transformations and represents them in the form given by system (2), is presented.

5.1 An algorithm to find a linear parameterization of structural controllable monitors

The algorithm extends the one presented in Moody *et al.* (1996) finding a basis for the valid constraint transformations. In Moody and Antsaklis (2000) it is shown that a structure for the admissible constraints can be characterized working on the basis of the kernel of the matrix $[\boldsymbol{C}_u \quad \boldsymbol{I}]^T$; our algorithm provide a systematic procedure to do it. The advantage of such parameterization w.r.t. to the Moody's one shown in prop. 4 is the fact that it ensures the structural controllability condition verification. In the next subsection the algorithm property to determine a maximal monitor based controllers family will be discussed. Without loss of generality, from now on we confine our attention to the case $n_c = 1$.

Algorithm
Input: $\boldsymbol{C}_u \in \mathbb{Z}^{m \times n_u}$, $\boldsymbol{l} \in \mathbb{Z}^{1 \times m}$.
Output $\boldsymbol{q} \in \mathbb{N}^{1 \times m}$, $x \in \mathbb{N}$, $z \in \mathbb{N}$, $\boldsymbol{P} \in \mathbb{N}^{z \times m}$, $\boldsymbol{A} \in \mathbb{Z}^{x \times z}$, $\boldsymbol{b} \in \mathbb{Z}^{x \times 1}$, $r_2 \in \mathbb{N}$, $infeasible : boolean$.

$\boldsymbol{q} := \boldsymbol{0}_{1 \times m}$; $r_2 = 1$; $x := 0$; $z := 0$;
if $[\mathbf{not}(\boldsymbol{l} \cdot \boldsymbol{C}_u \leq \boldsymbol{0})]$ **then**
(* Check if there is an arc from the monitor to an uncontrollable transition to move up *)
begin
$r_2 := 1$; $\boldsymbol{f}(1, \cdot) := \boldsymbol{0}_{1 \times n_u}$; $\boldsymbol{N} := \boldsymbol{0}_{1 \times n_u}$; $\boldsymbol{n} := \boldsymbol{l} \cdot \boldsymbol{C}_u$;
Let IR be a cellular array where $IR(s)$ is the set of row indexes of negative elements in the column $\boldsymbol{C}_u(., t_s)$ and $ir(s)$ its cardinality;
(* $IR(s)$ represents the set of input places of t_s *)
repeat
begin
$loop := false$;
Let $F = \{i \mid \boldsymbol{n}(i) > 0 \bigvee [\exists r, \boldsymbol{N}(r, i) > \boldsymbol{0}]\}$;
Choose $s \in F$; (* The algorithm chooses to move up the arc from the monitor to the uncontrollable transition, denoted by t_s *)
if $[\nexists r, \boldsymbol{f}(r, s) = 0]$ **then** $loop := true$;
(* $\boldsymbol{f}(r, s) = 1$ means that the uncontrollable transition t_s has yet been considered *)
if $[ir(s) \geq 1 \bigwedge \mathbf{not}(loop)]$ **then**
begin
Let β be the least common multiplier (l.c.m.) between $\boldsymbol{C}_u(p_i, t_s)$ for $i \in IR(s)$;
$[\boldsymbol{n} \quad \boldsymbol{q} \quad r_2] := \beta\, [\boldsymbol{n} \quad \boldsymbol{q} \quad r_2]$; $\boldsymbol{N} := \beta \boldsymbol{N}$;
if $[ir(s) = 1 \bigwedge \boldsymbol{N}(\cdot, s) \geq 0]$ **then make_null_$\boldsymbol{n}(s)$_and_$\boldsymbol{N}(\cdot, s)$**;
if $[ir(s) > 1 \bigvee (ir(s) = 1 \bigwedge \boldsymbol{N}(\cdot, s) \not\geq 0)]$ **then make_null_$\boldsymbol{n}(s)$_&_$\boldsymbol{N}(\cdot, s)$_with_symbols**;
end
else if $loop$ **then**
begin
$loop := 0$;
add_equation_to_make_null_$\boldsymbol{n}(s)$_and_$\boldsymbol{N}(\cdot, s)$;
if $[\boldsymbol{n}(s) > 0 \bigwedge (\boldsymbol{N}(\cdot, s) \leq \boldsymbol{0})]$ **then** $infeasible := true$;
if $[\boldsymbol{n}(s) = 0 \bigwedge \boldsymbol{A}(x, \cdot) \leq \boldsymbol{0}]$ **then check_feasibility**;
end
else $infeasible := true$;
end
until $[(\boldsymbol{n} \leq \boldsymbol{0} \bigwedge \boldsymbol{N} \leq \boldsymbol{0})$ **or** $infeasible]$
end

Fig. 4. Algorithm.

The algorithm (see figs. 4, 5, 6), works on the following table of integers: $\left[\begin{array}{c|c|c} \boldsymbol{C}_u & \boldsymbol{I}_{m \times m} & \boldsymbol{O}_{m \times 1} \\ \hline \boldsymbol{n} & \boldsymbol{q} & r_2 \end{array}\right]$, where at the initial step $\boldsymbol{n} = \boldsymbol{l}\boldsymbol{C}_u$, $\boldsymbol{q} = \boldsymbol{0}_{1 \times m}$, $r_2 = 1$ as in the algorithm of Moody. In fig. 7 it is reported an example to show how the algorithm works on the table of integers.

At each step, the algorithm goal is to "make null" a positive element of $\boldsymbol{n}$, the last row of the table, suppose $\boldsymbol{n}(t_s)$, by choosing as pivot a negative element in the s-th column of the table, suppose $\boldsymbol{C}_u(p_r, t_s)$ (see **procedure make_null_$\boldsymbol{n}(s)$_and_$\boldsymbol{N}(\cdot, s)$**). By adding the last row of the table multiplied by $-\boldsymbol{C}_u(p_r, t_s)$ with the r-th row multiplied by $\boldsymbol{n}(t_s)$ and replacing the last row with the one obtained from the addition, the result is achieved. If we denote as $(\boldsymbol{l}', k')$ the transformed constraint at the current step of the algorithm, we have that $\boldsymbol{l}' = r_2 \boldsymbol{l} + \boldsymbol{q}$, $k' = r_2(k + 1) - 1$ and $\boldsymbol{n} = \boldsymbol{l}' \boldsymbol{C}_u$. By this operation the component of vector $\boldsymbol{l}'$ relative to a place $p_r \in {}^\bullet t_s$ has been augmented by a positive quantity. A monitor place derived from $\boldsymbol{l}'$ has no more an output arc directed to transition t_s but an output arc directed to a transition $t \in {}^\bullet p_r$ as a result of the constraint transformation.

```
procedure make_null_n(s)_and_N(·,s)_adding_symbols
Input s, ir(s), IR(s), n, x, z, N, P, A, b, f.
Output n, x, z, N, P, A, b, f.
  begin
  (* if transition t_s has more than one input place,
  symbolic variables have to be introduced*)
  A(1 : z+1, z+1 : z+ir(s)) := -0_{(z+1)×ir(s)};
  x := x+1; k := 0; (* x is the algorithm symbolic steps counter *);
  f(z+1 : z+1+ir(s), :) := 0_{ir(s)×n_u};
  k := 0;
  for i ∈ IR(s) do
        begin
        k := k+1;
        A(x, z+k) := -C_u(p_i, t_s);
        (* an equation for the new symbolic parameter is introduced *)
        P(z+k, ·) := e_i;
        N(z+k, ·) := C_u(p_i, ·); N(z+k, s) := 0;
        f(z+1+k, i) := 1;
        end
  for i := 1 to z do
        if [N(i, s) > 0] then
          begin
          A(x, i) := -N(i, s);
          f(z+j, ·) := f(z+j, ·)&f(i, ·) ∀j = 1..ir(s)
          end
  if [n(s) > 0] then
        begin
        b(x) := n(s);
        n(s) := 0;
        end
  z := z+ir(s) (* z is the symbolic variable counter *);
  end

procedure make_null_n(s)_and_N(·,s)
Input s, IR(s), β, n, q, z, N, P, f
Output n, q, N, P, f
  begin
  if [n(s) > 0] then
        begin
        q(IR(s)) := q(IR(s)) + n(s)/β;
        n := n + n(s)C_u(IR(s), .)/β;
        f(1, s) := 1;
        end
  for i = 1 to z do
  if [N(i, s) > 0] then
        begin
        P(i, IR(s)) := P(i, IR(s)) + N(i, s);
        N(i, ·) := N(i, ·) + N(i, s)C_u(IR(s), ·)/β;
        f(i+1, s) := 1;
        end
  end
```

Fig. 5. Algorithm procedures.

When more than a negative element is present in the s-th column of the matrix $\boldsymbol{C}_u$, different solutions are obtained by choosing as pivot each of them (see **procedure make_null_$\boldsymbol{n}(s)$_&_$\boldsymbol{N}(\cdot, s)$_with_symbols**) or a linear combination of them. Let β be the least common multiplier between negative elements in $\boldsymbol{C}_u(\cdot, t_s)$ and $IR(s)$ the set of their row indexes. To consider all possible solutions, a symbolic variable α_i, supposed to be natural valued, is introduced for each negative element of $\boldsymbol{C}_u(\cdot, t_s)$. In addition, if the symbolic variable α_i is associate to $\boldsymbol{C}_u(p_r, t_s)$, let us define $[\,\boldsymbol{N}(\alpha_i, \cdot) \quad \boldsymbol{P}(\alpha_i, \cdot)\,]$ as follows: $\boldsymbol{N}(\alpha_i, t) = \beta \boldsymbol{C}_u(p_r, t)$ if $t \neq t_s$, $\boldsymbol{N}(\alpha_i, t) = 0$ if $t = t_s$; $\boldsymbol{P}(\alpha_i, p) = 0$ if $p \neq p_r$, $\boldsymbol{P}(\alpha_i, p) = 1$ if $p = p_r$. In order to work with symbols the table is augmented as follows

$\left[\begin{array}{c|c|c} \boldsymbol{C}_u & \boldsymbol{I}_{m\times m} & \boldsymbol{O}_{m\times 1} \\ \hline \boldsymbol{n} & \boldsymbol{q} & r_2 \\ \hline \boldsymbol{N} & \boldsymbol{P} & \boldsymbol{O}_{z\times 1} \end{array}\right]$, where each row of $[\,\boldsymbol{N} \quad \boldsymbol{P} \quad \boldsymbol{O}_{z\times 1}\,]$ is relative to one of z symbolic variables. At each step $\boldsymbol{l}' = \boldsymbol{l} + \boldsymbol{q} + \boldsymbol{\alpha}^T\boldsymbol{P}$, $\boldsymbol{l}'\boldsymbol{C}_u = \boldsymbol{n} + \boldsymbol{\alpha}^T\boldsymbol{N}$ while the symbolic variables do not compare in the computation of k'. The pivot element is now given by negative elements of the s-th column multiplied by α_i. By imposing that the linear combination of pivot elements is equal to $\beta\boldsymbol{n}(t_s)$, it can be verified that $\boldsymbol{l}'\boldsymbol{C}_u(\cdot, t_s) = 0$. This equation is added to the system $\boldsymbol{A\alpha} = \boldsymbol{b}$ (see Step 1 in fig.7 where $s = 1$ and $IR(s) = \{2, 3\}$).

After symbolic variables have been introduced, the algorithm goal at each step is to "make null" all positive elements in a column of the portion of the table

$\left[\begin{array}{c} \boldsymbol{n} \\ \hline \boldsymbol{N} \end{array}\right]$, since $\boldsymbol{l}'\boldsymbol{C}_u = \boldsymbol{n} + \boldsymbol{\alpha}^T\boldsymbol{N}$ and symbolic variables are not negative. At each step a new equation could be added to the system $\boldsymbol{A\alpha} = \boldsymbol{b}$ since new symbolic variables may be introduced even if only one negative element is present in the s-th column of the matrix $\boldsymbol{C}_u$ (see Step 3 where $s = 2$ and Step 5 where $s = 3$ in fig.7). The algorithm stops when there is no positive element in $\boldsymbol{n}$ or $\boldsymbol{N}$, or when there is no negative element in a column of $\boldsymbol{C}_u$ to make null a positive element in $\boldsymbol{n}$ or $\boldsymbol{N}$, i.e. the control law is infeasible.

Other algorithms proposed in (Li and Wonham, 1994) based on the state equation or in (Moody *et al.*, 1996) based on making null positive elements of $\boldsymbol{l}\boldsymbol{C}_u$ to which our algorithm is inspired, requires that $\boldsymbol{C}_u$ is acyclic. Instead, our algorithm works in presence of cycles of uncontrollable transitions. At this aim a flag matrix $\boldsymbol{f}$ is introduced: $\boldsymbol{f}(r+1, s) = 1$ means that the r-th symbolic variable multiplied by a negative element of $\boldsymbol{C}_u(\cdot, t_s)$ has been already used to make zero a positive element of $\boldsymbol{N}(\cdot, t_s)$. The same occurs when $\boldsymbol{f}(1, s) = 1$ as for making null $\boldsymbol{n}(t_s)$. Thus, if an element of the vector $\boldsymbol{f}(\cdot, t_s)$ is equal to one and the algorithm tries to make null again $\boldsymbol{n}(t_s)$ or $\boldsymbol{N}(\cdot, t_s)$, it will enter in a loop. This fact can be interpreted on the net graph as the presence of an uncontrollable transition cycle to which t_s belongs. When a cycle of uncontrollable transitions is detected, the constraint can be transformed in a controllable one imposing with a new equation that the quantity $\boldsymbol{l}'\boldsymbol{C}_u(\cdot, t_s) = \boldsymbol{n}(t_s) + \sum_{i=1}^{z} \alpha_i \boldsymbol{N}(\alpha_i, t_s)$ is null (see **procedure add_equation_to_make_null_$\boldsymbol{n}(s)$_and_$\boldsymbol{N}(\cdot, s)$**). If $\boldsymbol{n}(s) > 0$ and $\boldsymbol{N}(\cdot, t_s) > \boldsymbol{0}$, the equation has not solution. If $\boldsymbol{n}(t_s) = 0$ by imposing $\boldsymbol{l}'\boldsymbol{C}_u(\cdot, t_s) = 0$, it may occur that all variables involved in this equation has to assume a null value and so they are eliminated and the equation is eliminated too (see **procedure delete_symbol_and_equation_x**). As a consequence of eliminating symbols, some equation of the system $\boldsymbol{A\alpha} = \boldsymbol{b}$ may no more have solution, thus a check is needed (see **procedure check_feasibility**). Consider the net system in fig. 8a without dashed arcs and places and the uncontrollable specification $\boldsymbol{m}(p_1) \le 1$. From the algorithm we obtain the controllable sub-specification $\boldsymbol{m}(p_1) + \boldsymbol{m}(p_2) \le 1$ corresponding to the monitor p_c.

```
procedure add_equation_to_make_null_n(s)_and_N(·,s)
Input s, n, x, z, N, A, b.
Output n, x, N, A, b.
  begin
  x := x + 1;
  for i := 1 to z do
        A(x,i) := −N(i,s); N(i,s) := 0
  b(x) := n(s); n(s) := 0
  end


procedure delete_symbol_and_equation_x
Input x, z, N, P, A, b, f.
Output x, z, N, P, A, b, f.
  begin
  i:=1;
  while i ≤ z
        begin
        if [A(x,i) < 0] then
          begin
          delete N(i,·), P(i,·), A(·,i), f(i+1,·); z := z − 1;
          end
        i:=i+1;
        end
  delete A(x,·), b(x); x := x − 1;
  end


procedure check_feasibility
Input x, z, N, P, A, b, f, infeasible.
Output x, z, N, P, A, b, f, infeasible.
  begin
  delete_symbol_and_equation_x
  if z := 0 then infeasible := true;
  else if z > 0 ⋀ then
          for i = 1 to x do
            begin
            if [b(s) > 0 ⋀ A(x,·) ≤ 0] then infeasible := true;
            if [b(s) < 0 ⋀ A(x,·) ≥ 0] then infeasible := true;
            end
  end
```

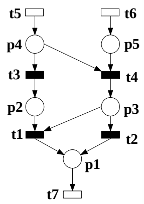
Fig. 6. Algorithm procedures for uncontrollable transition cycles.

The class of constraints $\Omega_1(\mathcal{L})$ is represented in the form

$$
\begin{aligned}
&\boldsymbol{l}' = \boldsymbol{r}_1 + r_2\boldsymbol{l}, \quad \boldsymbol{k}' = r_2(k+1) - 1 \\
&with \quad \boldsymbol{r}_1 = \boldsymbol{q} + \boldsymbol{\alpha}^T\boldsymbol{P}, \quad r_2 \in \mathbb{N} \\
&s.t. \begin{cases} \boldsymbol{A}\boldsymbol{\alpha} = \boldsymbol{b} \\ \boldsymbol{\alpha} \in \mathbb{N}^z \end{cases}
\end{aligned}
\tag{3}
$$

The algorithm provides not only maximal elements of $\Omega_1'(\mathcal{L})$, but a wider class of monitors. Consider the net system in fig. 8b and the uncontrollable GMEC $\mathcal{M}(\boldsymbol{l}, k) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{m}(p_3) \le 1\}$. Applying the algorithm of fig. 4 we obtain the transformed legal marking sets $\mathcal{M}(\boldsymbol{l}^1, k^1) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{m}(p_2) + \boldsymbol{m}(p_3) \le 1\}$ and $\mathcal{M}(\boldsymbol{l}^2, k^2) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{m}(p_1) + \boldsymbol{m}(p_2) + \boldsymbol{m}(p_3) \le 1\}$. Notice that $\mathcal{M}(\boldsymbol{l}^1, k^1) \supset \mathcal{M}(\boldsymbol{l}^2, k^2)$, and thus $\mathcal{M}(\boldsymbol{l}^1, k^1)$ is less restrictive.

Example 7 *Notice that the parameterization obtained from the algorithm can*



	t_1	t_2	t_3	t_4		
p_1	1	1	0	0		
p_2	−1	0	1	0		
p_3	−1	−1	0	1	$\boldsymbol{C}_u$	
p_4	0	0	−1	−1		
p_5	0	0	0	−1		
	1	1	0	0	$\boldsymbol{n} = \boldsymbol{l}\boldsymbol{C}_u$	$\boldsymbol{l} = [1 \quad 0 \quad 0 \quad 0 \quad 0]\,, k = 1$
*Step*1	0	1	0	0	$\boldsymbol{n}$	$\boldsymbol{l}' = [1 \quad \alpha_1 \quad \alpha_2 \quad 0 \quad 0]\,, k' = 1$
α_1	0	0	1	0	$\boldsymbol{N}$	$\alpha_1 + \alpha_2 = 1$
α_2	0	−1	0	1		$\boldsymbol{\alpha} \in \mathbb{N}^2$
*Step*2	0	1	0	0	$\boldsymbol{n}$	$\boldsymbol{l}' = [1 \quad \alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \alpha_4]\,, k' = 1$
α_1	0	0	1	0		$\alpha_1 + \alpha_2 = 1$
α_2	0	−1	0	0		$\alpha_3 + \alpha_4 = \alpha_2$
α_3	0	0	−1	0	$\boldsymbol{N}$	$\boldsymbol{\alpha} \in \mathbb{N}^4$
α_4	0	0	0	0		
*Step*3	0	0	0	0	$\boldsymbol{n}$	$\boldsymbol{l}' = [1 \quad \alpha_1 \quad \alpha_2 + \alpha_5 \quad \alpha_3 \quad \alpha_4]\,, k' = 1$
α_1	0	0	1	0		$\alpha_1 + \alpha_2 = 1$
α_2	0	0	0	0		$\alpha_3 + \alpha_4 = \alpha_2$
α_3	0	0	−1	0	$\boldsymbol{N}$	$\alpha_5 = 1 - \alpha_2$
α_4	0	0	0	0		$\boldsymbol{\alpha} \in \mathbb{N}^5$
α_5	−1	0	0	1		
*Step*4	0	0	0	0	$\boldsymbol{n}$	$\boldsymbol{l}' = [1 \quad \alpha_1 \quad \alpha_2 + \alpha 5 \quad \alpha_3 + \alpha 6 \quad \alpha_4 + \alpha_7]\,, k' = 1$
α_1	0	0	1	0		$\alpha_1 + \alpha_2 = 1$
α_2	0	0	0	0		$\alpha_3 + \alpha_4 = \alpha_2$
α_3	0	0	−1	0		$\alpha_5 = 1 - \alpha_2$
α_4	0	0	0	0	$\boldsymbol{N}$	$\alpha_6 + \alpha_7 = \alpha_5$
α_5	−1	0	0	0		$\boldsymbol{\alpha} \in \mathbb{N}^7$
α_6	0	0	−1	0		
α_7	0	0	0	0		
*Step*5	0	0	0	0	$\boldsymbol{n}$	$\boldsymbol{l}' = [1 \quad \alpha_1 \quad \alpha_2 + \alpha 5 \quad \alpha_3 + \alpha 6 + \alpha 8 \quad \alpha_4 + \alpha_7]\,, k' = 1$
α_1	0	0	0	0		$\alpha_1 + \alpha_2 = 1$
α_2	0	0	0	0		$\alpha_3 + \alpha_4 = \alpha_2$
α_3	0	0	0	0		$\alpha_5 = 1 - \alpha_2$
α_4	0	0	0	0	$\boldsymbol{N}$	$\alpha_6 + \alpha_7 = \alpha_5$
α_5	−1	0	0	0		$\alpha_8 = \alpha_1 - \alpha_3 - \alpha_6$
α_6	0	0	0	0		$\boldsymbol{\alpha} \in \mathbb{N}^8$
α_7	0	0	0	0		
α_8	0	0	0	−1		

Fig. 7. Algorithm steps to transform the uncontrollable specification $\boldsymbol{m}(p_1) \leq 1$ for the net in figure. The transformed constraint obtained at each step is reported.

be further simplified by reducing the number of parameters, by a simple triangularization of matrix $\boldsymbol{A}$ in the system (3). In the case of fig. 7 we have

$$a) \begin{cases} \alpha_2 = 1 - \alpha_1 \\ \alpha_4 = 1 - \alpha_1 - \alpha_3 \\ \alpha_5 = \alpha_1 \\ \alpha_6 = \alpha_1 - \alpha_7 \\ \alpha_8 = \alpha_7 - \alpha_3 \\ \boldsymbol{\alpha} \in \mathbb{N}^8 \end{cases} \Rightarrow b) \begin{cases} 1 - \delta_1 \geq 0 \\ 1 - \delta_1 - \delta_2 \geq 0 \\ \delta_1 - \delta_3 \geq 0 \\ \delta_3 - \delta_2 \geq 0 \\ \boldsymbol{\delta} \in \mathbb{N}^3 \end{cases} \tag{4}$$

We can write $\boldsymbol{l}' = [1 \quad \delta_1 \quad 1 \quad \delta_1 \quad 1 - \delta_1 - \delta_2 + \delta_3]$ and $k' = 1$ subject to the constraints (4b). By comparing (4a) and (4b) we observe that the number of symbols has been decreased from 8 to 3. In general case we can reduce the number of symbols to $z' \leq z$ and the system (3) to

$$
\begin{aligned}
&\boldsymbol{l}' = \boldsymbol{r}_1 + r_2\boldsymbol{l}, \quad \boldsymbol{k}' = r_2(k+1) - 1\\
&with \quad \boldsymbol{r}_1 = \boldsymbol{q} + \boldsymbol{\delta}^T\boldsymbol{P}, \quad r_2 \in \mathbb{N}\\
&s.t. \begin{cases} \boldsymbol{A}'\boldsymbol{\delta} \geq \boldsymbol{b}' \\ \boldsymbol{\delta} \in \mathbb{N}^{z'} \end{cases}
\end{aligned}
$$

5.2 Algorithm property

In this section we show that the parameterization computed via the algorithm of fig.4 under the assumption that the considered GMEC have positive weights has the property to include all maximal elements of the class $\Omega_1(\mathcal{L})$.

Assumption 1 *From now on we consider the legal marking sets having the form* $\mathcal{L} = \mathcal{M}(\boldsymbol{l}, k)$ *where* $\boldsymbol{l} \in \mathbb{N}^{1\times m}, k \in \mathbb{N}$.

Let us define the $\boldsymbol{l}$-uncontrollable subnet in order to characterize maximal elements in $\Omega_1(\mathcal{L})$ since afterwards we show that no place outside the $\boldsymbol{l}$-uncontrollable subnet has to be present in the transformed constraint, because this would lead to an useless restriction of the legal marking set under control.

Definition 8 *Let consider a PN* N *with* $T_c \neq \emptyset$ *and a GMEC* $(\boldsymbol{l}, k)$*. We define the* $\boldsymbol{l}$-uncontrollable subnet *of* N*, denoted as* $N_l = \langle P_l, T_l, \mathbf{Pre}_l, \mathbf{Post}_l\rangle$*, the subnet obtained from* N *eliminating every transition that does not belong to the set* $T_l = \{t | \exists p \in Q_l, \pi(t,p) \neq \emptyset\}$ *where* $\pi(t,p) = \{t_1p_1...t_kp_k | \forall i, t_i \in T_u, t_1 = t, p_k = p, Pre(p_i, t_{i+1}) > 0 \text{ and } Post(p_i, t_i) > 0\}$ *and every place that does not belong to the set* $P_l = Q_l \cup \{p | \exists t \in T_l, p \in {}^\bullet t\}$*. In other words,* N_l *contains places in* Q_l *and all uncontrollable transitions (and their input places) from which there exists a directed path with only uncontrollable transitions that leads to a place in* Q_l*.*

Consider the net system in fig. 3 and the control specification $\mathcal{L} = \mathcal{M}(\boldsymbol{l}, k) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{m}(p_1) \leq 1\}$. In this case $Q_l = \{p_1\}$, $P_l = \{p_1, p_2, p_3, p_4, p_5\}$, $T_l = \{t_2t_3t_4\}$.

Given a vector $\boldsymbol{x} : P \to \mathbb{N}$, the vector $\boldsymbol{x}' = \boldsymbol{x}|P_l$ is defined as follows: $\boldsymbol{x}'(p) = \boldsymbol{x}(p)$ if $p \in P_l$ and $\boldsymbol{x}'(p) = 0$ otherwise.

Now we define a class of constraint, denoted as $\Omega_1'(\mathcal{L})$, whose support is contained in P_l. Afterwards, we will show that maximal elements of $\Omega_1(\mathcal{L})$ are included in $\Omega_1'(\mathcal{L})$ and so the transformed constraints can be computed by working on a net that has a smaller size, without losing the optimality of our supervisory control problem solution.

Definition 9 *Let define the set* $\Omega'_1(\mathcal{L}) = \{\mathcal{K} \subseteq \mathcal{L} \mid \exists \boldsymbol{l'} \in \mathbb{N}^{1\times m}, k' \in \mathbb{N} : \mathcal{K} = \mathcal{M}(\boldsymbol{l'}, k'), \mathcal{K}$ *is structurally controllable and* $Q_{\boldsymbol{l'}} \subseteq P_l\}$ *of* structurally controllable *and* expressed by a linear inequality *subsets of* $\mathcal{L}$ having a support contained in the set P_l.

Lemma 10 *Given* $(\boldsymbol{l}, k)$ *then* $\forall(\boldsymbol{l'}, k')$ *such that* $\mathcal{M}(\boldsymbol{l'}, k') \subseteq \mathcal{M}(\boldsymbol{l}, k)$, *it holds* $Q_l \subseteq Q_{l'}$.

PROOF. By contradiction, assume $\exists \overline{p} \in Q_l \setminus Q_{l'}$. Let $\boldsymbol{m}$ be a marking so defined: $\boldsymbol{m}(\overline{p}) = \lceil \frac{k+1}{l(\overline{p})} \rceil$, $\boldsymbol{m}(p) = 0$ if $p \neq \overline{p}$. It follows $\boldsymbol{l'm} = 0 \leq k'$, $\boldsymbol{lm} = \boldsymbol{l}(\overline{p})\boldsymbol{m}(\overline{p}) \geq k+1$, thus $\boldsymbol{m} \in \mathcal{M}(\boldsymbol{l'}, k') \setminus \mathcal{M}(\boldsymbol{l}, k)$ violating the assumption. □

The following result shows that a constraint, that includes in its support a place that does not belong to the $\boldsymbol{l}$-uncontrollable subnet, is not maximal.

Proposition 11 *Let* $\mathcal{M}(\boldsymbol{l}, k)$ *be not structurally controllable and let* $\mathcal{M}(\boldsymbol{l'}, k')$ *be structurally controllable with* $\mathcal{M}(\boldsymbol{l'}, k') \subseteq \mathcal{M}(\boldsymbol{l}, k)$. *Let us define* $\boldsymbol{l''} = \boldsymbol{l'}|P_l$. *Then it holds:*
a) $\mathcal{M}(\boldsymbol{l''}, k')$ *is structurally controllable;*
b) $\mathcal{M}(\boldsymbol{l'}, k') \subseteq \mathcal{M}(\boldsymbol{l''}, k')$ *and in particular* $\mathcal{M}(\boldsymbol{l'}, k') \subsetneq \mathcal{M}(\boldsymbol{l''}, k')$ *if* $\boldsymbol{l'} \gneq \boldsymbol{l''}$;
c) $\mathcal{M}(\boldsymbol{l''}, k') \subseteq \mathcal{M}(\boldsymbol{l}, k)$.

PROOF. (a) First note that, by lemma 10, $Q_l \subseteq Q_{l'}$. Hence $Q_{l''} = Q_{l'} \cap P_l \supseteq Q_{l'} \cap Q_l = Q_l$ and $P_l \supseteq Q_{l''} \supseteq Q_l$. Assume by contradiction that (a) does not hold. Hence $\exists \tilde{t} \in T_u | \boldsymbol{l''}\boldsymbol{C}_u(\cdot, \tilde{t}) > 0 \Rightarrow \exists \tilde{p} \in Q_{l''} | \boldsymbol{C}_u(\tilde{p}, \tilde{t}) > 0 \Rightarrow \exists \tilde{p} \in Q_l | \boldsymbol{C}_u(\tilde{p}, \tilde{t}) > 0 \Rightarrow \tilde{t} \in T_l$. Thus $\forall p \in {}^\bullet\tilde{t}, p \in P_l$. It is possibile to write $\boldsymbol{l'}\boldsymbol{C}_u(\cdot, \tilde{t}) = \boldsymbol{l''}\boldsymbol{C}_u(\cdot, \tilde{t}) + \underbrace{\sum_{p \in \{\tilde{t}^\bullet \setminus {}^\bullet\tilde{t}\}, p \notin P_l} \boldsymbol{l'}(p)\boldsymbol{C}_u(p, \tilde{t})}_{\geq 0} \geq \boldsymbol{l''}\boldsymbol{C}_u(\cdot, \tilde{t}) \geq 0$ where $\boldsymbol{l''}\boldsymbol{C}_u(\cdot, \tilde{t}) = \sum_{p \in {}^\bullet\tilde{t}} \boldsymbol{l'}(p)\boldsymbol{C}_u(p, \tilde{t}) + \boldsymbol{l'}(\tilde{p})\boldsymbol{C}_u(\tilde{p}, \tilde{t}) + \sum_{p \in \{\tilde{t}^\bullet \setminus {}^\bullet\tilde{t}\}, p \neq \tilde{p}} \boldsymbol{l'}(p)\boldsymbol{C}_u(p, \tilde{t})$, clearly contradicting that $\mathcal{M}(\boldsymbol{l''}, k')$ is structurally controllable.

(b) For a generic $\boldsymbol{m}$ it holds $\boldsymbol{l'm} \leq k' \Rightarrow \boldsymbol{l''m} + \underbrace{(\boldsymbol{l'} - \boldsymbol{l''})\boldsymbol{m}}_{\geq 0} \leq k' \Rightarrow \boldsymbol{l''m} \leq k'$, so $\mathcal{M}(\boldsymbol{l'}, k') \subseteq \mathcal{M}(\boldsymbol{l''}, k')$. The strict containment follows if we consider $\boldsymbol{m}$ such that $(\boldsymbol{l'} - \boldsymbol{l''})\boldsymbol{m} > 0$.

(c) By contradiction, assume $\exists \boldsymbol{m} \in \mathcal{M}(\boldsymbol{l''}, k') \setminus \mathcal{M}(\boldsymbol{l}, k)$. Let us consider $\tilde{\boldsymbol{m}} = \boldsymbol{m}|P_l$, obviously $\tilde{\boldsymbol{m}} \notin \mathcal{M}(\boldsymbol{l}, k)$ by observing that $\boldsymbol{l}\tilde{\boldsymbol{m}} = \boldsymbol{lm} > k$, since $P_l \supseteq Q_l$. We can write $\boldsymbol{l'}\tilde{\boldsymbol{m}} = \boldsymbol{l''}\tilde{\boldsymbol{m}} \leq k' \Rightarrow \tilde{\boldsymbol{m}} \in \mathcal{M}(\boldsymbol{l'}, k') \setminus \mathcal{M}(\boldsymbol{l}, k)$ contradicting the hypothesis that $\mathcal{M}(\boldsymbol{l'}, k') \subseteq \mathcal{M}(\boldsymbol{l}, k)$. □

Hence we have the following corollary whose proof follows from proposition 11.

Corollary 12 *An element of $\Omega_1(\mathcal{L})$ is maximal only if it is an element of $\Omega_1'(\mathcal{L})$.*

Next two lemmas will be used to show the property of the algorithm to determine a parameterization of all the constraints that belong to the class $\Omega_1'(\mathcal{L})$.

Lemma 13 *Assume that $\mathcal{M}(\boldsymbol{l}, k)$ is not structurally controllable. It is possible to find a structurally controllable set $\mathcal{M}(\boldsymbol{l}', k)$ such that $\mathcal{M}(\boldsymbol{l}', k) \subseteq \mathcal{M}(\boldsymbol{l}, k)$ only if $\forall t \in T_l \mid \boldsymbol{l}\boldsymbol{C}(\cdot, t) > 0,\ \exists p \in {}^\bullet t, p \in Q_{l'}$.*

PROOF. By contradiction, let suppose that $\exists \tilde{t} \in T_l \mid \boldsymbol{l}\boldsymbol{C}(\cdot, \tilde{t}) > 0$ but $\boldsymbol{l}'\boldsymbol{C}(\cdot, \tilde{t}) \leq 0$ and $\forall p \in {}^\bullet\tilde{t}, p \notin Q_{l'}$ but $\mathcal{M}(\boldsymbol{l}', k) \subseteq \mathcal{M}(\boldsymbol{l}, k)$. We have that $\boldsymbol{l}\boldsymbol{C}(\cdot, \tilde{t}) - \boldsymbol{l}'\boldsymbol{C}(\cdot, \tilde{t}) > 0 \Rightarrow (\boldsymbol{l} - \boldsymbol{l}')\boldsymbol{C}(\cdot, \tilde{t}) \geq 0$.

It is possible to write $(\boldsymbol{l}-\boldsymbol{l}')\boldsymbol{C}(\cdot, \tilde{t}) = \sum_{p\in{}^\bullet\tilde{t}}(\boldsymbol{l}(p)-\boldsymbol{l}'(p))\boldsymbol{C}(p, \tilde{t}) + \sum_{p\in\{\tilde{t}^\bullet\setminus{}^\bullet\tilde{t}\}}(\boldsymbol{l}(p)-\boldsymbol{l}'(p))\boldsymbol{C}(p, \tilde{t}) = \underbrace{\sum_{p\in{}^\bullet\tilde{t}} \boldsymbol{l}(p)\boldsymbol{C}(p, \tilde{t})}_{\leq 0} + \sum_{p\in\{\tilde{t}^\bullet\setminus{}^\bullet\tilde{t}\}}(\boldsymbol{l}(p) - \boldsymbol{l}'(p))\boldsymbol{C}(p, \tilde{t}) \geq 0$.

Thus $\exists \overline{p} \in \tilde{t}^\bullet \setminus {}^\bullet\tilde{t} \mid \boldsymbol{l}(\overline{p}) > \boldsymbol{l}'(\overline{p})$. Then let $\boldsymbol{m}$ be a marking so defined: $\boldsymbol{m}(\overline{p}) = \lceil \frac{k+1}{l(\overline{p})} \rceil$, $\boldsymbol{m}(p) = 0$ if $p \neq \overline{p}$. It follows $\boldsymbol{l}'\boldsymbol{m} = 0 \leq k'$, $\boldsymbol{l}\boldsymbol{m} = \boldsymbol{l}(\overline{p})\boldsymbol{m}(\overline{p}) \geq k + 1$, hence $\boldsymbol{m} \in \mathcal{M}(\boldsymbol{l}', k') \setminus \mathcal{M}(\boldsymbol{l}, k)$, thus violating the assumption. □

The previous lemma shows that in order to transform an uncontrollable constraint into a structurally controllable one it is necessary to include in the support of the transformed constraint at least one input place of each uncontrollable transition having arcs directed to a place that belongs to the support of the uncontrollable constraint. The next lemma shows how to do this in a maximally permissive way.

Lemma 14 *Let $\mathcal{M}(\boldsymbol{l}, k)$ be not structurally controllable; by definition $\exists t \in T_l \mid \boldsymbol{l}\boldsymbol{C}(\cdot, t) > 0$. The largest subset $\mathcal{M}(\boldsymbol{l}', k')$ such that $\boldsymbol{l}'\boldsymbol{C}(\cdot, t) \leq 0$, $\mathcal{M}(\boldsymbol{l}', k') \subseteq \mathcal{M}(\boldsymbol{l}, k)$ and $p \in Q_{l'}$ with $p \in {}^\bullet t$, is given by $\boldsymbol{l}'(p) = -\boldsymbol{C}(p, t)\boldsymbol{l}(p) + \boldsymbol{l}\boldsymbol{C}(\cdot, t)$ and $\boldsymbol{l}'(p') = -\boldsymbol{C}(p, t)\boldsymbol{l}(p'), \forall p' \neq p$, $k' = -\boldsymbol{C}(p, t)(k+1) - 1$. It results $\boldsymbol{l}'\boldsymbol{C}(\cdot, t) = 0$.*

PROOF. First note, as shown in (Moody and Antsaklis, 2000), that if we let $\bar{\boldsymbol{l}} = b\boldsymbol{l}$ and $\bar{k} = b(k+1) - 1$ with $b \in \mathbb{N}$, it holds $\mathcal{M}(\boldsymbol{l}, k) = \mathcal{M}(\bar{\boldsymbol{l}}, \bar{k})$. In our case $b = -\boldsymbol{C}(p, t) \in \mathbb{N}$, since $\boldsymbol{C}(p, t) < 0$ being $p \in {}^\bullet t$. Suppose now to transform $(\bar{\boldsymbol{l}}, \bar{k})$ into $(\bar{\boldsymbol{l}}', \bar{k})$ by choosing $\bar{\boldsymbol{l}}'(p) = \bar{\boldsymbol{l}}(p) + \boldsymbol{l}\boldsymbol{C}(\cdot, t)$ and $\bar{\boldsymbol{l}}'(p') = \bar{\boldsymbol{l}}(p'), \forall p' \neq p$. Since $\boldsymbol{C}(p, t) < 0$ it results $\bar{\boldsymbol{l}}' \geq \bar{\boldsymbol{l}}$ that implies $\mathcal{M}(\bar{\boldsymbol{l}}', \bar{k}) \subseteq \mathcal{M}(\bar{\boldsymbol{l}}, \bar{k})$. It is immediate to verify that $\bar{\boldsymbol{l}}'\boldsymbol{C}(\cdot,t) = 0$. Any transformation $\tilde{\boldsymbol{l}}$ such that $\tilde{\boldsymbol{l}}\boldsymbol{C}(\cdot,t) < 0$ is a valid one but it is immediate to verify that $\tilde{\boldsymbol{l}} \gneq \bar{\boldsymbol{l}}'$ and consequently $\mathcal{M}(\bar{\boldsymbol{l}}',\bar{k}) \supsetneq \mathcal{M}(\tilde{\boldsymbol{l}},\bar{k})$ (see proposition 11b). $\square$

We can now state the following result whose proof comes from the fact that the algorithm makes an exhaustive search on the net N_l to make null any positive element in the vector $\boldsymbol{l}\boldsymbol{C}_l$ according to lemma 13 and lemma 14.

Proposition 15 *Under the Assumption 1 the Algorithm of fig. 4 determines a parameterization of constraints class that contains all the elements in* $\Omega_1'(\mathcal{L})$.

Lemma 13 and lemma 14 are correct even when the assumption 1 does not hold (the proof can be extended to take this case into account); we conjecture that also corollary 12 holds if the assumption 1 is removed.

5.3 Algorithm complexity

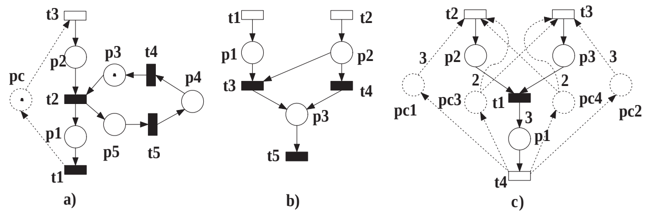


Fig. 8. a) a net with an uncontrollable circuit $p_5t_5p_4t_4p_3t_2$. b) Transition t_2 controls the firing of t_3 and t_4, while t_1 only the firing of t_3; if p_1 is included in the transformed constraint support, p_2 has to be necessary included too. c) A P/T net with four different monitors to enforce the uncontrollable specification $\boldsymbol{m}(p_1) \leq 3$.

At each step of the algorithm an addition of a number of rows (at most $max_{t\in T_u}|^\bullet t|$) of an integer table eventually pre-multiplied by an integer number is performed. In absence of symbolic variables the algorithm complexity (we mean the number of steps) would be in the worst case $O(t)$, with $t \in T_u$, because it is not possible to consider more than one time an uncontrollable transition. In presence of symbolic variables, if we denote by z their number, we have an algorithm complexity equal to $z\,O(t)$ in the worst case.

As for the number of symbolic variables, they are introduced in presence of transitions having more than one input place, i.e. in presence of synchronization (an output arc of a monitor place is directed to an uncontrollable synchronization transition): for such transition, a number of symbolic variables equal to the their input places is introduced. Let us introduce $s(t): T \rightarrow \mathbb{N}$, $s(t) = 0$ if $\mid {^\bullet t} \mid = 1$, otherwise $s(t) = \mid {^\bullet t} \mid$. Under the hypothesis that for each place p in the set $Q_{\boldsymbol{l}}$ there is no transition shared by two directed paths starting from a controllable transition and ending with an uncontrollable transition in the set $^\bullet p$ we have $z = \sum_{t\in T_u} s(t)$. This is the case of the net in fig. 3 with the control specification $\mathcal{L} = \mathcal{M}(\boldsymbol{l}, k) = \{\boldsymbol{m} \in \mathbb{N}^m \mid \boldsymbol{m}(p_1) \leq 1\}$. Here $Q_{\boldsymbol{l}} = \{p_1\}$, there are two control paths ($t_5p_4t_3p_2t_2$ and $t_6p_5t_4p_3t_2$) but no place is shared and $s = [\,0 \quad 2 \quad 0 \quad 0 \quad 0 \quad 0\,]$, thus we have two symbolic variables.

When a place p in the set $Q_{\boldsymbol{l}}$ has more that one input transition, if a transition t such that $s(t) > 0$ belongs to paths ending with different transitions of the set $^\bullet p$, then a number of $s(t)$ symbolic variables for each path has to be introduced. Let $r(t)$ be the cardinality of the subset of transitions $^\bullet p$ with $p \in Q_{\boldsymbol{l}}$ to which t is connected by a directed path, $z = \sum_{t\in T_u} s(t)r(t)$.

Finally, when places shared by more paths are present, it may be necessary to introduce a symbolic variable for each shared place. We conclude $z \leq \sum_{t\in T_u} s(t)r(t)+ \mid P_u \mid$ in the worst case.

Consider the example in fig. 7. By definition $s = [\,2 \quad 0 \quad 0 \quad 2 \quad 0 \quad 0 \quad 0\,]$, $r = [\,1 \quad 1 \quad 1 \quad 2 \quad 1 \quad 1 \quad 1\,]$ since t_4 is directly connected to t_1 and t_2 and $^\bullet p^1 = \{t_1, t_2\}$. In addition, p_4 is shared by the directed paths $t_5p_4t_3p_2t_1$, $t_5p_4t_4p_3t_1$ and p_3 is shared by the directed paths $t_5p_4t_4p_3t_1$, $t_6p_5t_4p_3t_2$. Thus a number of 8 symbolic variables is expected according to the result obtained (see fig. 7). Notice that the number of monitors may be greater than the number of symbolic variables. Consider the net in fig. 8c and the constraint $m(p_1) \leq 3$. The algorithm transforms the constraint into $m(p_1)+\alpha_1 m(p_2)+\alpha_2 m(p_3) \leq 3$ with $\alpha_1+\alpha_2 = 3$. Four different monitor can be derived: p_{c1},p_{c2} p_{c3}, p_{c4} (corresponding to $(\alpha_1, \alpha_2) = (3, 0)$,$(\alpha_1, \alpha_2) = (0, 3)$, $(\alpha_1, \alpha_2) = (2, 1)$, $(\alpha_1, \alpha_2) = (1, 2)$ respectively), while $z =\mid {}^\bullet t_1 \mid= 2$.

6 Conclusions and future works

In this paper the problem to compute a monitor based controller to enforce a GMEC on a plant net in presence of uncontrollable transitions has been focused on. Firstly, we have shown that there is not an optimal solution to this problem. Then, we have proposed an algorithm to compute structurally controllable monitors enforcing a given GMEC and to present them in form of a unique control net incidence matrix that depends linearly on the value of the parameters subject to a linear equations system.

In this way the logical constraint enforcement may be considered as the first step of a DES control design, and thus it is useful, when a supremal solution at this level does not exists, to present all the possible solutions in a form suitable to solve an optimization problem at the next design step.

References

Ghaffari, A., N. Rezg and Xiaolan Xie (2003). Design of a live and maximally permissive petri net controller using the theory of region. *IEEE Trans. On Robotics and Automation* **19**(1), 137–142.

Giua, A., F. DiCesare and M. Silva (1992). Generalized mutual exclusion constraints on nets with uncontrollable transitions. *1992 IEEE Int. Conf. on Systems, Man, and Cybernetics (Chigago, Illinois)* pp. 974–979.

Holloway, L.E., B.H. Krogh and A. Giua (1997). A survey of Petri nets methods for controlled discrete event systems. *Discrete Event Dynamic Systems: Theory and Applications* **7**(2), 151–190.

Krogh, B.H. and L.E. Holloway (1991). Synthesis of feedback control logic for discrete maufacturing systems. *Automatica* **27**(4), 641–651.

Li, Y. and W.M. Wonham (1994). Control of vector discrete-event systems II - controller synthesis. *IEEE Trans. on Automatic Control* **39**(3), 512–531.

Moody, J. and P. Antsaklis (2000). Petri net supervisors for DES with uncontrollable and unobservable transitions. *IEEE Trans. on Aut. Control* **45**(3), 462–476.

Moody, J.O., M.D. Lemmon and P. Antsaklis (1996). Supervisory control of Petri nets with uncontrollable/unobservable transitions. 35^{th} *IEEE Conf. on Decision and Control, (Kobe, Japan)* **4**, 4433–4438.

Ramadge, P.J. and W.M. Wonham (1989). The control of discrete-event systems. *Proc. of IEEE* **77**(1), 81–98.

Stremersch, G. (2000). Linear algebraic design of supervisors for partially observed Petri nets. *IFAC Conf. on Control System Design. Bratislava, Slovak Republic.*

Zhou, M.C. and F. DiCesare (1993). *Petri Net Synthesis for Discrete Event Control of Manufacturing Systems.* Kluwer Ac.