On the Enforcement of a Class of Nonlinear Constraints on Petri Nets

YuFeng Chen, ZhiWu Li, Kamel Barkaoui, and Alessandro Giua *

April 14, 2015

Abstract

This paper focuses on the enforcement of nonlinear constraints in Petri nets. First, a supervisory structure is proposed for a nonlinear constraint. The proposed structure consists of added places and transitions. It controls the transitions in the net to be controlled only but does not change its states since there is no arc between the added transitions and the places in the original net. Second, an integer linear programming model is proposed to transform a nonlinear constraint to a minimal number of conjunctive linear constraints that have the same control performance as the nonlinear one. By using a place invariant based method, the obtained linear constraints can be easily enforced by a set of control places. The control places consist to a supervisor that can enforce the given nonlinear constraint. On condition that the admissible markings space of a nonlinear constraint is non-convex, another integer linear programming model is developed to obtain a minimal number of constraints whose disjunctions are equivalent to the nonlinear constraint. Finally, a number of examples are provided to demonstrate the proposed approach.

Published as:
Y.F. Chen, Z.W. Li, K. Barkaoui, A. Giua, "On the Enforcement of a Class of Nonlinear Constraints on Petri Nets," Automatica, Vol. 55, pp. 116-124, 2015. DOI: 10.1016/j.automatica.2015.02.014.

*YuFeng Chen is with the School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China, and also with Cedric Lab and Computer Science Department, Conservatoire National des Arts et Métiers, Paris 75141, France (e-mail: yfchen@mail.xidian.edu.cn). ZhiWu Li is with Institute of Systems Engineering, Macau University of Science and Technology, Macau, School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China, and Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia (e-mail: zhwli@xidian.edu.cn). Kamel Barkaoui is with Cedric Lab and Computer Science Department, Conservatoire National des Arts et Métiers, Paris 75141, France (e-mail: kamel.barkaoui@cnam.fr). Alessandro Giua is with Aix Marseille Université, CNRS, ENSAM, Université de Toulon, LSIS UMR 7296, 13397, Marseille, France, DIEE, University of Cagliari, 09123 Cagliari, Italy, and School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China (emails: giua@diee.unica.it and alessandro.giua@lsis.org).

1 Introduction

Petri nets [28] are a powerful tool to model and analyze discrete event systems (DESs). They have been widely used for deadlock control, scheduling and planning, and performance evaluation in a variety of resource allocation systems [2, 22, 23, 39]. Supervisory control is a suitable mechanism to enforce external constraints on a system to be controlled. In the framework of Petri nets, a supervisor that enforces supervisory control specifications is often represented by a set of control places.

Constraints associated with reachable states in a DES are a typical and important control specification in supervisory control theory of DESs. Many specifications can be converted into linear constraints. For example, deadlock problems in Petri nets are usually dealt with by finding a set of constraints, with respect to the markings, that can prevent the system from reaching deadlock states [3, 8, 20, 21]. Most control requirements in system control design can be directly represented by a set of constraints.

Generally, there are two classes of constraints in Petri nets: linear and nonlinear. Linear constraints, also called generalized mutual exclusion constraints (GMECs) [12, 27], play an important role in the development of supervisors for a system modeled by Petri nets. Many efforts have been done to enforce a GMEC by constructing a place invariant (PI) [1, 7, 38]. The PI-based approach is well-established and widely used by researchers and engineers. In [38], Yamalidou *et al.* study a variety of GMECs and design control places to enforce them by constructing PIs. In [15, 16], Iordache *et al.* present an approach to the implementation of disjunctive GMECs. The work in [14] provides a good survey on the design of control places by PI based methods. Up to now, a lot of work has been done to deal with deadlocks by Petri nets [10, 11, 13, 17, 18, 19, 24, 37]. In fact, almost all of them compute control places by PIs [9]. Another line to deal with deadlocks in DESs is based on finite state automata, as shown in [30, 31, 33]. In this work, we focus on the enforcement of nonlinear constraints on Petri nets.

In [34, 35], Uzam and Zhou provide an iterative method to design liveness-enforcing supervisors of Petri nets. They divide the reachability graph of a Petri net model into two parts: a live-zone (LZ) and a deadlock-zone (DZ), where the LZ contains all legal markings and the DZ includes all the illegal markings from which no legal marking is reachable. First, they compute the set of first-met bad markings (FBMs) of a net model. An FBM is an illegal marking that represents the very first entry from the LZ to the DZ. At each iteration, an FBM is singled out and a control place is computed to forbid it. The process cannot terminate until all FBMs are forbidden. Then, the controlled net is live since it cannot enter the DZ any more. The method is intuitive but cannot lead to an optimal supervisor in general. In our previous work [4], we improve Uzam and Zhou's results by proposing a method to obtain a maximally permissive supervisor. In [5, 6], the control places are computed by solving an integer linear programming problem (ILPP) that makes all legal markings reachable but all FBMs unreachable. Meanwhile, the objective function can minimize the number of the control places.

However, not all specifications can be represented as GMECs. In some cases, the specifications require to enforce a nonlinear constraint on a net model. For GMECs, the control places can be designed by constructing PIs. However, to the best of our knowledge, no work is reported to compute a supervisor by following the clue of handling GMECs if the constraints are nonlinear since we cannot directly construct PIs for the nonlinear constraints. This work focuses on the enforcement of nonlinear constraints. A supervisory structure is developed. It splits a transition in an original net model into a set of transitions. The proposed supervisor can also make all markings in the admissible-zone reachable and all markings in the forbidden-zone unreachable. The proposed approach is applicable to bounded Petri net models.

The rest of the paper is organized as follows. In Section 2, some basics of Petri nets are recalled. Section 3 reports the concepts and properties of nonlinear constraints. Section 4 provides a supervisory structure to implement a nonlinear constraint. Meanwhile, a number of examples are provided to illustrate the performance of the supervisory structure. Finally, conclusions are reached in Section 5.

2 Preliminaries

This section recalls the basics of Petri nets [28] and generalized mutual exclusion constraints (GMECs) [12].

2.1 Petri nets

A Petri net is a four-tuple $N = (P, T, F, W)$ where P and T are finite and non-empty sets. P is a set of places and T is a set of transitions with $P \cap T = \emptyset$. $F \subseteq (P \times T) \cup (T \times P)$ is a flow relation of the net, represented by arcs with arrows from places to transitions or from transitions to places. $W : (P \times T) \cup (T \times P) \to \mathbb{N}$ is a mapping that assigns a weight to an arc: $W(x, y) > 0$ if $(x, y) \in F$, and $W(x, y) = 0$, otherwise, where $(x, y) \in (P \times T) \cup (T \times P)$ and $\mathbb{N}$ is the set of non-negative integers. ${}^\bullet x = \{y \in P \cup T \mid (y, x) \in F\}$ is called the preset of x and $x^\bullet = \{y \in P \cup T \mid (x, y) \in F\}$ is called the postset of x. A marking is a mapping $M : P \to \mathbb{N}$. $M(p)$ denotes the number of tokens in place p. The pair (N, M_0) is called a marked Petri net or a net system. A net is pure (self-loop free) if $\forall (x, y) \in (P \times T) \cup (T \times P)$, $W(x, y) > 0$ implies $W(y, x) = 0$. The incidence matrix $[N]$ of a net N is a $|P| \times |T|$ integer matrix with $[N](p, t) = W(t, p) - W(p, t)$.

A transition $t \in T$ is enabled at marking M if $\forall p \in {}^\bullet t$, $M(p) \geq W(p, t)$. This fact is denoted as $M[t\rangle$. Once an enabled transition t fires, it yields a new marking M', denoted as $M[t\rangle M'$, where $M'(p) = M(p) - W(p, t) + W(t, p)$. The set of reachable markings of net N with initial marking M_0 is denoted by $R(N, M_0)$. It can be graphically expressed by a reachability graph, denoted as $G(N, M_0)$. It is a directed graph whose nodes are markings in $R(N, M_0)$ and arcs are labeled by the fired transitions.

Let (N, M_0) be a net system with $N = (P, T, F, W)$. A transition $t \in T$ is live if $\forall M \in R(N, M_0)$, $\exists M' \in R(N, M)$, $M'[t\rangle$. (N, M_0) is live if $\forall t \in T$, t is live. It is dead if $\nexists t \in T$, $M_0[t\rangle$.

2.2 Generalized Mutual Exclusion Constraint

A GMEC [12] is a control requirement that limits a weighted sum of tokens contained in a subset of places. Let $[N]$ be the incidence matrix of a plant with n places and m transitions. A GMEC can be expressed as:

$$\sum_{i=1}^{n} w_i \cdot \mu_i \leq k \tag{1}$$

where μ_i denotes the number of tokens in place p_i at any reachable marking, and w_i and k are non-negative integers. Eq.(1) can be represented as a vector form, i.e.,

$$\vec{w}^T \cdot \vec{\mu} \leq k \tag{2}$$

where $\vec{w}$ is a weight vector of nonnegative integers with $\vec{w}(i) = w_i$, $\vec{\mu}$ is a vector of nonnegative integers with $\vec{\mu}(i) = \mu_i$ and k is a positive integer. A GMEC is usually denoted as $(\vec{w}, k)$.

By introducing a non-negative slack variable μ_s, Eq.(2) becomes

$$\vec{w}^T \cdot \vec{\mu} + \mu_s = k \tag{3}$$

where μ_s represents the marking of control place p_s, generally called a monitor. The firing of a transition t modifies the tokens in p_s by a constant:

$$\Delta(t) = -\vec{w}^T \cdot [N](\bullet, t) \tag{4}$$

In fact, $\forall M_1, M_2 \in R(N, M_0)$ with $M_1 = M_2 + [N](\bullet, t)$, we have $\Delta(t) = M_1(p_s) - M_2(p_s)$. Thus, the incidence vector $[N_s]$ of p_s can be computed by:

$$[N_s] = -\vec{w}^T \cdot [N] \tag{5}$$

The initial marking $M_0(p_s)$ of p_s can be calculated as follows:

$$M_0(p_s) = k - \vec{w}^T \cdot M_0 \tag{6}$$

3 Generalizations of Arbitrary Marking Constraints

In this section, we present basic concepts of nonlinear constraints in Petri nets in the sense of reachability graph analysis. A constraint for a Petri net is in general a predicate with respect to the states (markings) of the Petri net. Let c be a constraint that restricts the tokens contained in a subset of places of a Petri net model (N, M_0). In this work, the constraints are only associated with markings while no firing vectors of transitions are considered.

Definition 1 *Let c be a constraint and $M \in R(N, M_0)$ a marking of a net (N, M_0). The function $F(c, M)$ is defined as $F(c, M) = 1$ if M satisfies c and $F(c, M) = 0$ otherwise.*

Given a constraint c, the reachable markings of a net are classified into two groups: admissible ones that satisfy c and inadmissible ones that do not satisfy c, as defined below:

Definition 2 *Let c be a constraint of a Petri net model (N, M_0). A marking $M \in R(N, M_0)$ is said to be admissible with respect to c if $F(c, M) = 1$. The set of admissible markings of c is denoted by $\mathcal{M}_c$. A reachable marking M of (N, M_0) is said to be inadmissible with respect to c if $F(c, M) = 0$. The set of inadmissible markings of c is denoted by $\mathcal{M}_{\overline{c}}$.*

Given a constraint for a Petri net model (N, M_0), we assume that its initial marking always satisfies the constraint. Then, the reachability graph of (N, M_0) can be classified into two parts: an admissible-zone (AZ) and a forbidden-zone (FZ). There may exist some admissible markings that cannot be reached from the initial marking through admissible markings only. In this case, these admissible markings cannot be reached if all inadmissible markings are forbidden, i.e., they should be included in the FZ though they are admissible. Hence, the AZ includes the maximal set of admissible markings of c, which are reachable from the initial marking without leaving $\mathcal{M}_c$, whose set is denoted as $\mathcal{M}_c^\star$, and the FZ contains all the other reachable markings, i.e., all the inadmissible markings of c and the admissible markings that cannot be reached without leaving the AZ, whose set is denoted as $\mathcal{M}_{\overline{c}}^\star$. It is obvious that $\mathcal{M}_c^\star \subseteq \mathcal{M}_c$ and $\mathcal{M}_{\overline{c}} \subseteq \mathcal{M}_{\overline{c}}^\star$. The partition of a reachability graph is demonstrated in Fig. 1.

A supervisor is *maximally permissive*, or said to be *optimal*, if it can always disable any transition whose firing leads to a marking in the FZ and does not disable any transition

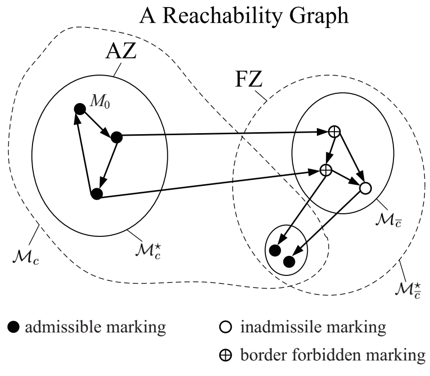

Figure 1: The AZ and the FZ in a reachability graph.

whose firing leads to a marking in the AZ. In this sense, a maximally permissive supervisor for a constraint c should keep all the admissible markings in the AZ of c and exclude the reachability of any marking in the FZ of c.

A border forbidden marking (BFM) of c is a marking in the FZ that is a direct successor of some marking in the AZ, as shown in Fig. 1. Mathematically, the set of BFMs, denoted by $\mathcal{M}_B$, is defined as follows:

Definition 3 *Let c be a constraint on (N, M_0). The set of BFMs is defined as* $\mathcal{M}_B = \{M | M \in \mathcal{M}_{\bar{c}}^\star, \exists M' \in \mathcal{M}_c^\star, \exists t \in T, \text{ s.t. } M'[t\rangle M\}$.

If all BFMs cannot be reached, their successors cannot be reached. Thus, there is no need to compute the whole reachability graph of a net system.

4 Design of Supervisory Structures for Nonlinear Constraints

In this section, we define a new class of constraints that are inspired by GMECs [12] but not linear. We propose a supervisory structure to implement a nonlinear constraint, which can optimally enforce it, i.e., all admissible markings in the AZ of the constraint are reachable.

4.1 Synthesis of an Optimal Supervisory Structure

In this section, we develop an approach to design a supervisor for a class of nonlinear constraints, namely an additive separable function, as defined below.

Definition 4 *An additive separable constraint c involves the sum of a number of functions $f_i(\mu_i)$ $(i \in \{1, 2, \ldots, n\})$ and a constant β, formally,*

$$f_1(\mu_1) + f_2(\mu_2) + f_3(\mu_3) + \ldots + f_n(\mu_n) \leq \beta \tag{7}$$

where $f_i(\mu_i)$ is a nonlinear function of μ_i and μ_i denotes the marking of p_i, $i \in \{1, 2, \ldots, n\}$.

For instance, $f_1(\mu_1) = \mu_1 \cdot \mu_1$ is a nonlinear function of μ_1. The support $||c||$ of an additive separable constraint c is defined as the set of places p_i such that $f_i(\mu_i)$ is not the zero function, i.e., $||c|| = \{p_i | f_i(\mu_i) \neq 0\}$. An additive separable constraint can be transformed into an equality by introducing a non-negative slack variable μ_s (the marking of control place p_s), as presented below:

$$f_1(\mu_1) + f_2(\mu_2) + f_3(\mu_3) + \ldots + f_n(\mu_n) + \mu_s = \beta \tag{8}$$

In such a case the firing of a transition t at a marking M modifies the slack variable μ_s of a quantity that depends on the marking M:

$$\Delta(t, M) = f(M) - f(M + [N](\bullet, t)) \tag{9}$$

where $f(\vec{\mu}) = f_1(\mu_1) + f_2(\mu_2) + f_3(\mu_3) + \ldots + f_n(\mu_n)$. Eq.(9) can be rewritten as

$$\Delta(t, M) = \sum_{i=1}^{n} [f_i(M(p_i)) - f_i(M(p_i) + [N](p_i, t))] \tag{10}$$

Property 1 *Let t be a transition and $\mathbb{N}_{\Delta_t} = \{\Delta(t, M)|\ M \in R(N, M_0)\}$. Then, $|\mathbb{N}_{\Delta_t}|$ (the cardinality of $\mathbb{N}_{\Delta_t}$) is finite.*

Proof: It can be easily obtained by the fact that $|R(N, M_0)|$ (the cardinality of $R(N, M_0)$) is finite. ■

In fact, $\Delta(t, M)$ has no relation with the marking of a place p_i if $[N](p_i, t) = 0$ or $p_i \notin ||c||$. Hence, Eq.(10) can be simplified as

$$\Delta(t, M) = \sum_{p_i \in (^\bullet t \cup t^\bullet) \cap ||c||} [f_i(M(p_i)) - f_i(M(p_i) + [N](p_i, t))] \tag{11}$$

Eq.(11) motivates us to transform a transition t into a set of transitions to represent the different modified quantities of control place p_s. By Eq.(11), $\Delta(t, M)$ is a sum of the token modifications $[f_i(M(p_i)) - f_i(M(p_i) + [N](p_i, t))]$, where $p_i \in (^\bullet t \cup t^\bullet) \cap ||c||$. Hence, we can design a supervisory structure for each nonlinear function $f_i(\mu_i)$ respectively and then combine them together to enforce the nonlinear constraint. Without loss of generality, let us design a supervisory structure for $f_1(\mu_1)$. For the sake of brevity, let $g(\mu_2, \mu_3, \ldots, \mu_n) = f_2(\mu_2) + f_3(\mu_3) + \ldots + f_n(\mu_n)$. Then, Eq.(8) can be written as

$$f_1(\mu_1) + g(\mu_2, \mu_3, \ldots, \mu_n) + \mu_s = \beta \tag{12}$$

In order to enforce the nonlinear constraint, each of the input and output transitions of the places involved in the nonlinear constraints is replaced by a set of transitions. Next, we show details of the design of the supervisory structure. We consider the supervisory structure for $f_1(\mu_1)$ in Eq.(7). An algorithm is presented as follows.

Algorithm 1 *Design of a supervisory structure for a nonlinear function*
Input: A bounded Petri net (N, M_0) and a nonlinear function $f_1(\mu_1)$
Output: A supervisory structure for $f_1(\mu_1)$
1) Let K_{p_1} be the upper bound of p_1[1] and $\mathbb{P}^z_{K_{p_1}} = \{(x, y)|\forall x, y \leq K_{p_1}, y - x = z\}$. $\forall t_j \in {}^\bullet p_1 \cup p_1^\bullet$, a set of transitions $\widehat{T}^z_j$ is designed to replace t_j, where $\widehat{T}^z_j = \{t^{x-y}_j|\forall (x, y) \in \mathbb{P}^z_{K_{p_1}}, z = [N](p_1, t_j)\}$.
2) $\forall p_k \in P/\{p_1\}, \forall t^{x-y}_j \in \widehat{T}^z_j$, add arcs: $W(p_k, t^{x-y}_j) = W(p_k, t_j)$ and *$W(t^{x-y}_j, p_k) = W(t_j, p_k)$. Remove the transitions in ${}^\bullet p_1 \cup p_1^\bullet$.*
3) Add a place $\bar{o}_1$ with $M_0(\bar{o}_1) = K_{p_1} - M_0(p_1)$, $W(\bar{o}_1, t^{x-y}_j) = K_{p_1} - x, \forall t^{x-y}_j \in \widehat{T}^z_j$. Add arcs: $W(p_1, t^{x-y}_j) = x, \forall t^{x-y}_j \in \widehat{T}^z_j$. / $\bar{o}_1$ is called a complementary place of p_1. */*

[1]Since the complexity of the supervisory structure increases with the upper bound of p_1, we make K_{p_1} as small as possible. For example, if the capacity of p_1 decided by the original net structure is 3 but the tokens in p_1 is limited to be no more than 2 by a given constraint, then K_{p_1} should be the smaller one, i.e., 2.

4) Add a control place p_s with $M_0(p_s) = \beta$. $\forall t_j^{x-y} \in \widehat{T}_1$, the arcs between p_s and t_j^{x-y} are defined as $[N_s](t_j^{x-y}) = f_1(x) - f_1(y), \forall t_j^{x-y} \in \widehat{T}_j^z$, where $[N_s]$ is the incidence vector of p_s.
5) Output the obtained supervisory structure.
6) End.

In Step 1, for each transition t_j in the preset or the postset of p_1, a set of transitions $\widehat{T}_j^z(z = [N](p_1, t_j))$ is designed to replace t_j.

Step 2 ensures that without considering p_1, $\forall t_j^{x-y} \in \widehat{T}_j^z$, t_j^{x-y} has the same enabled condition as t_j since transition t_j^{x-y} is used to replace t_j in the original net.

By Step 3, t_j^{x-y} can be enabled at a marking M only if $M(p_1) = x$. Once t_j^{x-y} fires, the number of tokens in p_1 should be y and the number of tokens in $\bar{o}_1$ is $K_{p_1} - y$.

Finally, Step 4 designs a control p_s to ensure that the marking of p_s satisfies Eq.(8).

We summarize the design of the improved supervisory structure in Table 1, where the first column shows a place p, the second and the third columns indicate the weights on arcs (p, t_j^{x-y}) and (t_j^{x-y}, p), respectively, and the last column represents the initial marking of p.

Table 1: Supervisory structure for $f_1(\mu_1)$ in nonlinear constraint Eq.(7)

p	$W(p, t_j^{x-y})$ $t_j^{x-y} \in \widehat{T}_j^z$, $t_j \in {}^\bullet p_1 \cup p_1^\bullet$	$W(t_j^{x-y}, p)$ $t_j^{x-y} \in \widehat{T}_j^z$, $t_j \in {}^\bullet p_1 \cup p_1^\bullet$	$M_0(p)$
$p_k \in P/p_1$	$W(p_k, t_j)$	$W(t_j, p_k)$	$M_0(p_k)$
p_1	x	y	$M_0(p_1)$
$\bar{o}_1$	$K_{p_1} - x$	$K_{p_1} - y$	$K_{p_1} - M_0(p_1)$
p_s	$\begin{cases} f_1(y) - f_1(x) & \text{if } f_1(y) > f_1(x) \\ 0 & \text{otherwise} \end{cases}$	$\begin{cases} f_1(x) - f_1(y) & \text{if } f_1(x) > f_1(y) \\ 0 & \text{otherwise} \end{cases}$	β

Example 1 *A simple constraint, Eq.(13), is used to illustrate the proposed approach.*

$$\mu_1 \cdot \mu_1 + \mu_2 \leq 4 \tag{13}$$

By introducing a non-negative slack variable μ_s, the inequality constraint can be transformed into an equality as follows:

$$\mu_1 \cdot \mu_1 + \mu_2 + \mu_s = 4 \tag{14}$$

where μ_s represents the marking of control place p_s. Suppose that the net to be controlled is shown in Fig. 2[2].

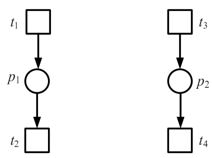

Figure 2: The subnet generated by $\{p_1, p_2\} \cup {}^\bullet p_1 \cup p_1^\bullet \cup {}^\bullet p_2 \cup p_2^\bullet$.

It can be seen that place p_1 is unbounded in the original net but we can obtain its upper bound by Eq.(13), i.e., $K_{p_1} = 2$. Then, we can design the supervisor as shown in Fig.

[2]Note that we only show the subnet generated by $\{p_1, p_2\} \cup {}^\bullet p_1 \cup p_1^\bullet \cup {}^\bullet p_2 \cup p_2^\bullet$ since Eq.(13) is concerned with the tokens in p_1 and p_2 only.

Table 2: Admissible markings for Eq.(13)

i	$M_i(p_1)$	$M_i(p_2)$	$M_i(p_s)$
0	0	0	4
1	0	1	3
2	0	2	2
3	0	3	1
4	0	4	0
5	1	0	3
6	1	1	2
7	1	2	1
8	1	3	0
9	2	0	0

3. There are four added transitions: t_1^{0-1}, t_1^{1-2}, t_2^{1-0}, and t_2^{2-1}, where t_1 and t_2 in the original net model are replaced by $\{t_1^{0-1}, t_1^{1-2}\}$ and $\{t_2^{1-0}, t_2^{2-1}\}$, respectively. It can be seen that a transition t_j^{x-y} ($j \in \{1,2\}$) is enabled at a marking M if $M(p_1) = x$ and $M(p_s) \geq W(p_s, t_j^{x-y})$. Once t_j^{x-y} fires at M, the marking of p_s changes to $M(p_s) - (f_1(y) - f_1(x))$. The reachability graph of the controlled net in Fig. 3 is shown in Fig. 4. We can verify that the controlled net is live with 10 reachable markings as shown in Table 2. That is to say, the proposed supervisor can implement the nonlinear constraint and make all admissible markings reachable. ■

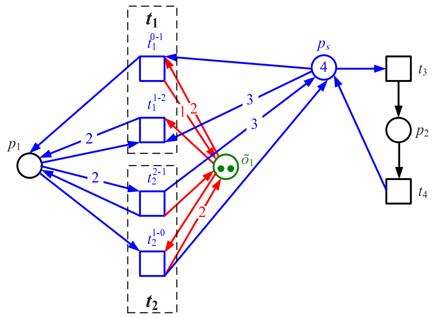

Figure 3: The supervisory structure for $\mu_1 \cdot \mu_1 + \mu_2 \leq 4$.

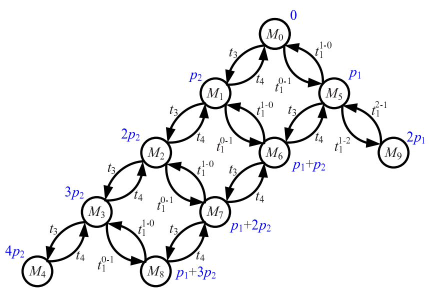

Figure 4: The rechability graph of the controlled net in Fig. 3.

Theorem 1 *Let M and M' be two markings in a plant net model (N, M_0) with $M'(p_1) = M(p_1) + [N](p_1, t_j)$ and $M'(p_i) = M(p_i)$, $i \in \{2, 3, \ldots, n\}$, where $[N]$ is the incidence matrix. Suppose that M is a reachable marking that satisfies Eq.(12), then the proposed supervisory structure due to Algorithm 1 can ensure that*
1) M' is reachable from M and satisfies the PI equality Eq.(12) if M' satisfies Eq.(7); and
2) M' is unreachable if M' does not satisfy Eq.(7).

Proof: First, we consider Case 1). Since M satisfies Eq.(12), we have $f_1(M(p_1)) + g(M) + M(p_s) = \beta$, where $g(M) = f_2(M(p_2)) + f_3(M(p_3)) + \ldots + f_n(M(p_n))$. Since M' satisfies Eq.(7), $f_1(M'(p_1)) + g(M') \leq \beta$ is true, where $g(M') = f_2(M'(p_2)) + f_3(M'(p_3)) + \ldots + f_n(M'(p_n))$. In this case, we have $M(p_s) = \beta - f_1(M(p_1)) - g(M) \geq f_1(M'(p_1)) + g(M') - f_1(M(p_1)) - g(M') = f_1(M'(p_1)) - f_1(M(p_1))$. Next, we consider two subcases 1.a) $f_1(x) - f_1(y) < 0$ and 1.b) $f_1(x) - f_1(y) > 0$.

For Case 1.a), consider p_1 and the added place $\overline{o}_1$. The added transition t_j^{x-y} with $x = M(p_1)$ and $y = M'(p_1)$ can fire at M owing to $M(p_s) \geq f_1(M'(p_1)) - f_1(M(p_1))$ and $W(p_s, t_j^{x-y}) = f_1(y) - f_1(x)$. Once t_j^{x-y} fires, we obtain a new marking M' with $M'(p_s) = M(p_s) - (f_1(y) - f_1(x))$. By $M'(p_i) = M(p_i)$, $i \in \{2, 3, \ldots, n\}$, $g(M') = g(M)$ holds. Therefore, we have $f_1(M'(p_1)) + g(M') + M'(p_s) = \beta$, i.e., M' is reachable from M and satisfies the PI equality Eq.(12).

Similarly, for Case 1.b), since t_j is enabled at M, t_j^{x-y} is enabled at M with $x = M(p_1)$ and $y = M'(p_1)$. Thanks to $W(t_j^{x-y}, p_s) = f_1(x) - f_1(y)$, once t_j^{x-y} fires, it yields a marking M', i.e., M' is reachable with $M'(p_s) = M(p_s) + f_1(y) - f_1(x)$. By $M'(p_i) = M(p_i)$, $i \in \{2, 3, \ldots, n\}$, $g(M') = g(M)$ is true. Hence, we have $f_1(M'(p_1)) + g(M') + M'(p_s) = \beta$, i.e., M' is reachable and satisfies the PI equality Eq.(12).

Next, we consider Case 2). By Eq.(12), we have $M(p_s) = \beta - f_1(M(p_1)) - g(M)$. Since M' does not satisfy Eq.(7), $\beta < f_1(M'(p_1)) + g(M')$ is true. Thus, we have $M(p_s) < f_1(M'(p_1)) + g(M') - f_1(M(p_1)) - g(M) = f_1(M'(p_1)) - f_1(M(p_1))$, leading to the fact that the added transition t_j^{x-y} with $x = M(p_1)$ and $y = M'(p_1)$ is disabled by p_s. As a result, M' is unreachable. ■

Theorem 1 indicates that the proposed supervisory structure due to Algorithm 1 can implement the nonlinear function $f_1(\mu_1)$. In this case, we can design such a supervisory structure for each nonlinear function $f_i(\mu_i)$ ($i \in \{1, 2 \ldots, n\}$) in Eq.(7) and merge them into a supervisor by the shared places p_s with $M_0(p_s) = \beta$. Then, the obtained supervisor can implement the nonlinear constraint Eq.(7). In the following, an algorithm is presented to merge the supervisory structure for each nonlinear function in Eq.(7).

Algorithm 2 *Design of a supervisory structure for an additive separable constraint*
Input: A bounded Petri net (N, M_0) *and an additive separable constraint* c
Output: A supervisory structure for c
1) Add a control place p_s *with* $M_0(p_s) = \beta - f(M_0)$.
2) ***foreach*** $\{p_i \in ||c||\}$ ***do*** {

Add a complementary place $\overline{o}_i$ *of* p_i *with* $M_0(\overline{o}_i) = K_{p_i} - M_0(p_i)$, *where* K_{p_i} *is the upper bound of* p_i.

foreach $\{t_j \in {}^\bullet p_i \cup p_i^\bullet\}$ ***do*** {

A set of transitions $\widehat{T}_j^z$ *is designed to replace* t_j, *where* $\widehat{T}_j^z = \{t_j^{x-y} | \forall (x, y) \in \mathbb{P}_{K_{p_i}}^z, z = [N](p_i, t_j)\}$.

$\forall t_j^{x-y} \in \widehat{T}_j^z$, *add arcs* $W(\overline{o}_i, t_j^{x-y}) = K_{p_i} - x$ *and* $W(p_i, t_j^{x-y}) = x$.

$\forall p_k \in P/\{p_i\}, \forall t_j^{x-y} \in \widehat{T}_j^z$, *add arcs* $W(p_k, t_j^{x-y}) = W(p_k, t_j)$ and $W(t_j^{x-y}, p_k) = W(t_j, p_k)$.

$\forall t_j^{x-y} \in \widehat{T}_j^z$, *the arcs between* p_s *and* t_j^{x-y} *are defined as* $[N_s](t_j^{x-y}) = [N_s](t_j) + f_i(x) - f_i(y), \forall t_j^{x-y} \in \widehat{T}_j^z$, *where* $[N_s]$ *is the incidence vector of* p_s.

Remove the transition t_j.

}

}

3) Output the obtained supervisory structure.
4) End.

Proposition 1 *Let (N^α, M_0^α) be a supervisor obtained by Algorithm 2. Then, the obtained supervisor can implement the nonlinear constraint Eq.(7).*

Proof: Let M and M' be two markings in a plant net model (N, M_0) with $M' = M + [N](\bullet, t_j)$. Suppose that M is a reachable marking that satisfies Eq.(12). Then, we can prove that the proposed supervisory structure by Algorithm 2 can ensure that
1) M' is reachable from M and satisfies the PI equality Eq.(8) if M' satisfies Eq.(7); and
2) M' is unreachable if M' does not satisfy Eq.(7).

For Case 1), we first consider that the tokens in two places p_1 and p_2 are changed by firing t_j, i.e., $[N](p_1, t_j) \neq 0$ and $[N](p_2, t_j) \neq 0$. Let $x_1 = M(p_1)$, $x_2 = M(p_2)$, $y_1 = M'(p_1)$, and $y_2 = M'(p_2)$. If M' satisfies Eq.(7), then there exists a transition $t_j^{x_1x_2-y_1y_2}$ in the set of added transitions representing t_j. Similar to the proof of Theorem 1, $t_j^{x_1x_2-y_1y_2}$ is enabled at M only if $x_1 = M(p_1)$ and $x_2 = M(p_2)$. In this case, $t_j^{x_1x_2-y_1y_2}$ can be fired and once it fires, it yields the new marking M' with $M'(p_1) = y_1$ and $M'(p_2) = y_2$. It can also be verified that $[N_s](t_j^{x_1x_2-y_1y_2}) = f_1(x_1) - f_1(y_1) + f_2(x_2) - f_2(y_2)$, where $[N_s]$ is the incidence vector of the control place p_s. Then, we have $M'(p_s) = M(p_s) + [N_s](t_j^{x_1x_2-y_1y_2})$, i.e., $M'(p_s) = M(p_s) + f_1(x_1) - f_1(y_1) + f_2(x_2) - f_2(y_2)$. Since M satisfies Eq.(8), i.e., $f_1(x_1) + f_2(x_2) + \ldots + f_n(M(p_n) + M(p_s) = \beta$, we have $f_1(y_1) + f_2(y_2) + \ldots + f_n(M'(p_n) + M'(p_s) = f_1(y_1) + f_2(y_2) + \ldots + f_n(M'(p_n) + M(p_s) + f_1(x_1) - f_1(y_1) + f_2(x_2) - f_2(y_2) = f_1(x_1) + f_2(x_2) + \ldots + f_n(M(p_n) + M(p_s) = \beta$. Hence, M' satisfies the PI equality Eq.(8). Now, we can similarly prove that Case 1) holds if the tokens in more than two places are changed by firing t_j.

Next, we consider Case 2). Similarly, we first consider that the tokens in two places p_1 and p_2 are changed by firing t_j. By Eq.(12), we have $M(p_s) = \beta - f_1(M(p_1)) - f_2(M(p_2)) - \ldots - f_n(M(p_n))$. Since M' does not satisfy Eq.(7), $\beta < f_1(M'(p_1)) + f_2(M'(p_2)) + \ldots + f_n(M'(p_n))$ is true. Thus, we have $M(p_s) < f_1(M'(p_1)) + f_2(M'(p_2)) + \ldots + f_n(M'(p_n)) - f_1(M(p_1)) - f_2(M(p_2)) - \ldots - f_n(M(p_n)) = f_1(M'(p_1)) - f_1(M(p_1)) + f_1(M'(p_2)) - f_1(M(p_2))$, leading to the fact that the added transition $t_j^{x_1x_2-y_1y_2}$ with $x_1 = M(p_1)$, $x_2 = M(p_2)$, $y_1 = M'(p_1)$, and $y_2 = M'(p_2)$ is disabled by p_s. As a result, M' is unreachable. Now, we can similarly prove that Case 2) holds if the tokens in more than two places are changed by firing t_j.

Finally, we conclude that the supervisor obtained by Algorithm 2 can implement the nonlinear constraint Eq.(7). ■

4.2 Structural Complexity

In this section, we discuss the structural complexity of the proposed supervisor. The number of added places is $n+1$ since there are n complementary places $\overline{o}_i$ $(i = 1, 2, \ldots, n)$ and a control place p_s. Next, we discuss the number of the added transitions as follows.

1) First, we consider the case that a transition t_j modifies the marking of just one place p_i in the support of c. Then, t_j is replaced by $|\widehat{T}_j^{z_{i,j}}|(z_{i,j} = [N](p_i, t_j))$ transitions.

2) Second, we consider the case that a transition t_j modifies the marking of two places p_{i_1} and p_{i_2} in the support of c. Then, at the iteration step to design the supervisory structure for p_{i_1}, t_j is replaced by $|\widehat{T}_j^{z_{i_1,j}}|$ transitions, where $z_{i_1,j} = [N](p_{i_1}, t_j)$. At the iteration step to design the supervisory structure for p_{i_2}, for each newly added transition in $\widehat{T}_j^{z_{i_1,j}}$, it is replaced by $|\widehat{T}_j^{z_{i_2,j}}|$ transitions, where $z_{i_2,j} = [N](p_{i_2}, t_j)$. Therefore, the total number of added transitions to replace t_j is $|\widehat{T}_j^{z_{i_1,j}}| \cdot |\widehat{T}_j^{z_{i_2,j}}|$.

3) Third, we consider the case that a transition t_j modifies the marking of r places $p_{i_1}, p_{i_2}, \ldots, p_{i_r}$ in the support of c. Then, the total number of added transitions to replace t_j is $|\widehat{T}_j^{z_{i_1,j}}| \cdot |\widehat{T}_j^{z_{i_2,j}}| \cdot \ldots \cdot |\widehat{T}_j^{z_{i_r,j}}|$.

4) Let $T_r = \{t \in T || ({}^\bullet t \cup t^\bullet) \cap ||c||| = r\}$ be such a set of transitions that each transition in it modifies the marking of r places in the support of c, and denote the places in ${}^\bullet T_r \cap T_r^\bullet$ as $\{p_{i_1}, p_{i_2}, \ldots, p_{i_r}\}$. Then, the total number of added transitions is $\sum_{r=1}^{n} \sum_{t_j \in T_r} \prod_{k=1}^{r} |\widehat{T}_j^{z_{i_k,j}}|$.

According to the above discussions, it can be seen that the proposed method suffers from supervisory complexity problem if there are too many transitions that modify the marking of multiple places in $||c||$. The reason is that the proposed method is applicable to all additive separable constraints. In fact, the supervisory structure can be simply reduced for some special constraints. In the following, we provide two simple examples to demonstrate this point.

Example 2 *We consider the following example*

$$f_i(\mu_i) = \begin{cases} 0 & \text{if} \quad \mu_i \le a \\ \mu_i - a & \text{if} \quad \mu_i > a \end{cases} \tag{15}$$

where $1 \le a \le K_{p_i}$ and K_{p_i} is the upper bound of p_i. In Eq.(15), a transition t that modifies the marking of place p_i needs only to be split into two transitions: one that does not modify the marking of p_s and the other that modifies the marking of place p_s by one token. The corresponding supervisory structure is shown in Fig. 5, where t_1 is split into two transitions: $t_1^{<a}$ whose firing does not modify the marking of p_s at a marking M if $M(p_i) < a$ and $t_1^{\ge a}$ whose firing reduces the marking of place p_s by one token at a marking M if $M(p_i) \ge a$. Similarly, t_2 is split into two transitions: $t_2^{\le a}$ whose firing does not modify the marking of p_s at a marking M if $M(p_i) \le a$ and $t_2^{>a}$ whose firing increases the marking of place p_s by one token at a marking M if $M(p_i) > a$.

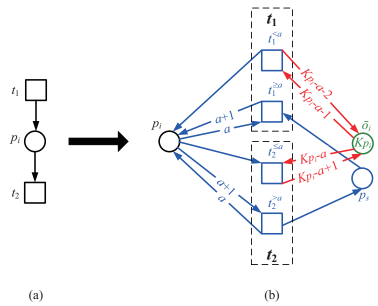

Figure 5: (a) A subnet for p_i and (b) the supervisory structure for Eq.(15).

Example 3 *A similar nonlinear function is shown in the following.*

$$f_i(\mu_i) = \begin{cases} 0 & \text{if} \quad \mu_i \le a \\ b & \text{if} \quad \mu_i > a \end{cases} \tag{16}$$

where $1 \le a \le K_{p_i}$ and K_{p_i} is the upper bound of p_i. In Eq.(16), a transition t that modifies the marking of place p_i needs only to be split into three transitions: two that do not modify the marking of p_s and one that modifies the marking of place p_s by b tokens. The corresponding supervisory structure is shown in Fig. 6, where t_1 is split into three transitions: $t_1^{<a}$ whose firing does not modify the marking of p_s at a marking M if $M(p_i) < a$, $t_1^{>a}$ whose firing does not modify the marking of p_s at a marking M if $M(p_i) > a$, and $t_1^{=a}$ whose firing reduces the marking of place p_s by b tokens at a marking M if $M(p_i) = a$. Similarly, t_2 is split into three transitions: $t_2^{<a+1}$ whose firing does

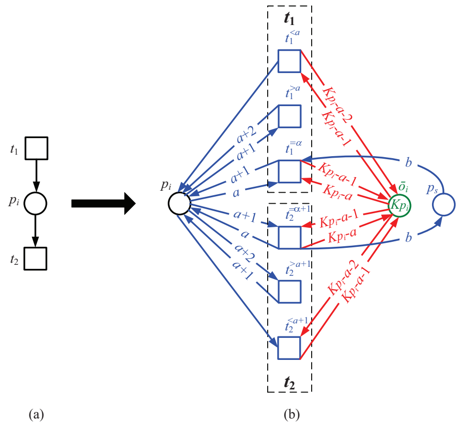

Figure 6: (a) A subnet for p_i and (b) the supervisory structure for Eq.(16).

not modify the marking of p_s *at a marking* M *if* $M(p_i) < a+1$, $t_2^{>a+1}$ *whose firing does not modify the marking of* p_s *at a marking* M *if* $M(p_i) > a+1$, *and* $t_2^{=a+1}$ *whose firing increases the marking of place* p_s *by* b *tokens at a marking* M *if* $M(p_i) = a+1$.

Examples 2 and 3 show that the supervisory structure for a nonlinear function $f_i(\mu_i)$ can be reduced if $f_i(\mu_i)$ can be divided into some linear parts. Then, for each linear part, a transition t that modifies the marking of place p_i should modify the marking in the control place p_s by a constant. Hence, we need only one transition to represent the linear part. As a result, the number of the added transitions to replace t is reduced. By the two examples, we can see that the proposed approach is particularly fit for piecewise linear functions since in that case the complexity of the resulting supervisor is much better than that suggested by the worst-case analysis.

4.3 An Example for the Proposed Supervisory Structure

In this section, an example is proposed to demonstrate the proposed supervisory structure.

Example 4 *We consider the missionaries and cannibals problem (MCP) [32]. It is a well-known toy problem in artificial intelligence, where it was used by Saul Amarel as an example of problem representation [36]. The MCP is briefly stated as follows. Three missionaries and three cannibals must cross a river by using a boat. The boat can carry at most two people. At each of the both banks, if there are missionaries present on the bank, their number must be no less than that of cannibals in the same bank. Otherwise, the cannibals would eat the missionaries. The boat cannot cross the river by itself if there is no people on board.*

We consider the control problem in the MCP. In each bank, the control strategy becomes:

$$\begin{cases} n_m \geq n_c & \text{if } n_c > 0 \\ n_m & \text{if } n_c = 0 \end{cases}$$

where $n_m \in \{0,1,2,3\}$ *and* $n_c \in \{0,1,2,3\}$ *represent the numbers of missionaries and cannibals in a bank, respectively. The control strategy can be transformed into an additive separable function as follows:*

$$f_1(n_m) + f_2(n_c) \leq 3 \tag{17}$$

where $f_1(n_m)$ *is a mapping from integers to integers as shown in Table 3 and* $f_2(n_c) = n_c$.

Table 3: Definition of $f_1(n_m)$

n_m	$f_1(n_m)$
0	0
1	2
2	1
3	0

We can verify that Eq.(17) can represent the control strategy in the MCP. Eq.(17) is an additive separable constraint that can be enforced by the method proposed in Section 4.1. The original Petri net modeling the MCP has six places and ten transitions, as shown in Fig. 7. Table 4 presents the meanings of transitions in the net model. It has 30 reachable markings in which 18 and 12 are admissible and inadmissible ones, respectively. Note that there are two admissible markings that can be reached by BFMs only, i.e., both are in the FZ. Thus, the two markings should not be reached once all BFMs are forbidden. By using the supervisor construction method proposed in Section 4.1, four transitions in the original net model are replaced by 16 transitions and two control places are added for the control strategies in the two banks. The incidence matrix $[N_c]$ of the controlled net model is shown in Table 5 where $[N_c](p, t) = a, -b$ indicates that there is a self-loop between p and t with $W(t, p) = a$ and $W(p, t) = b$. In the table, p_{s_1} with $M_0(p_{s_1}) = 0$ and p_{s_2} with $M_0(p_{s_2}) = 3$ are used to enforce constraints $f_1(\mu_1) + f_2(\mu_2) \leq 3$ and $f_1(\mu_3) + f_2(\mu_4) \leq 3$ for the control purpose in the left and right banks, respectively. Note that we do not design the complementary place of p_1 since p_3 is just the one with $M(p_1) + M(p_3) = 3$ for any reachable marking M. Similarly, p_1 is also the complementary place of p_3. The controlled net model has all the 16 admissible markings and no inadmissible marking is reachable. That is to say, the proposed method can optimally enforce the control strategies in the MCP. ■

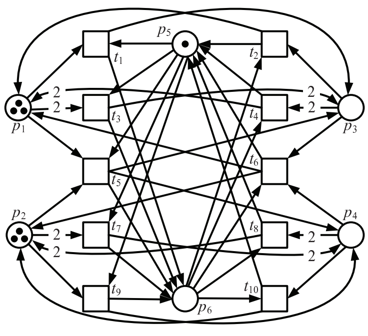

Figure 7: The Petri net model of MCP.

5 Conclusions

This paper deals with the enforcement of the nonlinear constraints on bounded Petri nets. A supervisory structure is presented to implement a class of nonlinear constraints, namely additive separable functions. The proposed method can directly design a supervisor given a nonlinear constraint. A number of examples are provided to demonstrate the proposed method. A future topic is to reduce the structural complexity of the proposed supervisors. Another future work is to extend this work to design Petri net supervisors to enforce nonlinear constraints for net models with uncontrollable transitions [29, 25, 26].

Table 4: Meanings of transitions in Fig. 7

t_i	meanings
t_1	boat carries a missionary to right bank
t_2	boat carries a missionary to left bank
t_3	boat carries two missionaries to right bank
t_4	boat carries two missionaries to left bank
t_5	boat carries a missionary and a cannibal to right bank
t_6	boat carries a missionary and a cannibal to left bank
t_7	boat carries two cannibals to right bank
t_8	boat carries two cannibals to left bank
t_9	boat carries a cannibal to right bank
t_{10}	boat carries a cannibal to left bank

Table 5: The incidence matrix $[N_c]$ of the controlled net model for MCP

$p\backslash t$	t_1^{3-2}	t_1^{2-1}	t_1^{1-0}	t_3^{3-1}	t_3^{2-0}	t_5^{3-2}	t_5^{2-1}	t_5^{1-0}	t_7	t_9	t_2^{0-1}	t_2^{1-2}	t_2^{2-3}	t_4^{0-2}	t_4^{1-3}	t_6^{0-1}	t_6^{1-2}	t_6^{2-3}	t_8	t_{10}
p_1	2, −3	1, −2	−1	1, −3	−2	2, −3	1, −2	−1	0	0	1	2, −1	3, −2	2	3, −1	1	2, −1	3, −2	0	0
p_2	0	0	0	0	0	−1	−1	−1	−2	−1	0	0	0	0	0	1	1	1	2	1
p_3	1	2, −1	3, −2	2	3, −1	1	2, −1	3, −2	0	0	2, −3	1, −2	−1	1, −3	−2	2, −3	1, −2	−1	0	0
p_4	0	0	0	0	0	1	1	1	2	1	0	0	0	0	0	−1	−1	−1	−2	−1
p_5	−1	−1	−1	−1	−1	−1	−1	−1	−1	−1	1	1	1	1	1	1	1	1	1	1
p_6	1	1	1	1	1	1	1	1	1	1	−1	−1	−1	−1	−1	−1	−1	−1	−1	−1
p_{s_1}	−1	−1	2	−2	1	0	0	3	2	1	−2	1	1	−1	2	−3	0	0	−2	−1
p_{s_2}	−2	1	1	−1	2	−3	0	0	−2	−1	−1	−1	2	−2	1	0	0	3	2	1

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61203038 and 61374068, the Fundamental Research Funds for the Central Universities under Grant No. JB140402, the Recruitment Program of Global Experts, and the Science and Technology Development Fund, MSAR, under Grant No. 066/2013/A2.

References

[1] Z. Banaszak and B. H. Krogh, "Deadlock avoidance in flexible manufacturing systems with concurrently competing process flows," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 6, pp. 724–734, 1990.

[2] K. Barkaoui and I. B. Abdallah, "A deadlock prevention method for a class of FMS," *IEEE International Conference on Systems, Man and Cybernetics, Intelligent Systems for the 21st Century,* vol. 5, pp. 4119–4124, 1995.

[3] K. Barkaoui, J. M. Couvreur, and K. Klai, "On the equivalence between liveness and deadlock-freeness in Petri nets," *Lecture Notes in Computer Science*, vol. 3536, pp 90–107, 2005.

[4] Y. F. Chen, Z. W. Li, M. Khalgui, and O. Mosbahi, "Design of a maximally permissive liveness-enforcing Petri net supervisor for flexible manufacturing systems," *IEEE Transactions on Automation Science and Engineering*, vol. 8, no.2, pp. 374–393, 2011.

[5] Y. F. Chen and Z. W. Li, "Design of a maximally permissive liveness-enforcing supervisor with a compressed supervisory structure for flexible manufacturing systems," *Automatica*, vol. 47, no. 5, pp. 1028–1034, 2011.

[6] Y. F. Chen and Z. W. Li, "On structural minimality of optimal supervisors for flexible manufacturing systems," *Automatica*, vol. 48, no. 10, pp. 2647–2656, 2012.

[7] Y. F. Chen, Z. W. Li, and M. C. Zhou, "Behaviorally optimal and structurally simple liveness-enforcing supervisors of flexible manufacturing systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 42, no. 3, pp. 615–629, 2012.

[8] Y. F. Chen and Z. W. Li, *Optimal Supervisory Control of Automated Manufacturing Systems*, New York: CRC Press, Taylor & Francis Group, 2013.

[9] Y. F. Chen, Z. W. Li, and M. C. Zhou, "Optimal supervisory control of flexible manufacturing systems by Petri nets: A set classification approach," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 2, pp. 549–563, 2014.

[10] Y. F. Chen, Z. W. Li, and K. Barkaoui, "New Petri net structure and its application to optimal supervisory control: Interval inhibitor arcs," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 10, pp. 1384–1400, 2014.

[11] A. Ghaffari, N. Rezg, and X. L. Xie, "Design of a live and maximally permissive Petri net controller using the theory of regions," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 1, pp. 137–142, 2003.

[12] A. Giua, F. DiCesare, and M. Silva, "Generalized mutual exclusion constraints on nets with uncontrollable transitions," in *Proceeding of IEEE International Conference on Systems, Man, and Cybernetics*, 1992, pp. 974–979

[13] Y. S. Huang, M. D. Jeng, X. L. Xie, and D. H. Chung, "Siphon-based deadlock prevention for flexible manufacturing systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 36, no. 6, pp. 1248-1256, 2006.

[14] M. V. Iordache and P. J. Antsaklis, "Supervision based on place invariants: A survey," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 16, no. 4, pp. 451–492, 2006.

[15] M. V. Iordache and P. J. Antsaklis, "A structural approach to the enforcement of language and disjunctive constraints," in *Proccesdings of American Control Conference*, 2005, pp. 3920–3925.

[16] M. V. Iordache and P. J. Antsaklis, "Petri net supervisors for disjunctive constraints," in *Proceedings of American Control Conference*, 2007, pp. 4951–4956.

[17] Z. W. Li and M. C. Zhou, "Elementary siphons of Petri nets and their application to deadlock prevention in flexible manufacturing systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 34, no. 1, pp. 38–51, 2004.

[18] Z. W. Li and M. C. Zhou, "Clarifications on the definitions of elementary siphons of Petri nets," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 36, no. 6, pp. 1227–1229, 2006.

[19] Z. W. Li and M. C. Zhou, "On siphon computation for deadlock control in a class of Petri nets," *IEEE Transactions on Systems, Man and Cybernetics, Part A-Systems and Humans*, vol. 38, no. 3, pp. 667–679, 2008.

[20] Z. W. Li and M. Zhao, "On controllability of dependent siphons for deadlock prevention in generalized Petri nets," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 38, no. 2, pp. 369–384, 2008.

[21] Z. W. Li and M. C. Zhou, *Deadlock Resolution in Automated Manufacturing Systems: A Novel Petri Net Approach*. Springer, London, 2009.

[22] Z. W. Li, G. Y. Liu, M-H. Hanisch, and M. C. Zhou, "Deadlock prevention based on structure reuse of Petri net supervisors for flexible manufacturing systems," *IEEE Transactions on Systems, Man and Cybernetics, Part A*, vol. 42, no. 1, pp. 178–191, 2012.

[23] Z. W. Li, N. Q. Wu, and M. C. Zhou, "Deadlock control of automated manufacturing systems based on Petri nets–A literature review," *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, vol. 42, no. 4, pp. 437–462, 2012.

[24] D. Liu, Z. W. Li, and M. C. Zhou, "Liveness of an extended S^3PR", *Automatica*, vol. 46, no. 6, pp. 1008–1018, 2010.

[25] J. L. Luo, H. Shao, K. Nonami, and F. J. Jin, "Maximally permissive supervisor synthesis based on a new constraint transformation method," *Automatica*, vol. 48, no. 6, pp. 1097–1101, 2012.

[26] Z. Y. Ma, Z.W. Li, and A. Giua, "Comments on maximally permissive supervisor synthesis based on a new constraint transformation method," *Automatica*, vol. 51, no. 1, pp. 131–134, 2015.

[27] Z. Y. Ma, Z. W. Li, and A. Giua, "Design of optimal Petri net controllers for disjunctive generalized mutual exclusion constraints," *IEEE Transactions on Automatic Control*, in press, DOI:10.1109/TAC.2015.2389313, 2015.

[28] T. Murata, "Petri nets: Properties, analysis and application," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.

[29] J. O. Moody and P. J. Antsaklis, "Petri net supervisors for DES with uncontrollable and unobservable transitions," *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 462–476, 2000.

[30] A. Nazeem, S. Reveliotis, Y. Wang and S. Lafortune, "Designing compact and maximally permissive deadlock avoidance policies for complex resource allocation systems through classification theory: The linear case," *IEEE Transactions on Automatic Control*, vol. 56, no. 8, pp. 1818–1833, 2011.

[31] A. Nazeem and S. Reveliotis, "Designing compact and maximally permissive deadlock avoidance policies for complex resource allocation systems through classification theory: The nonlinear case," *IEEE Transactions on Automatic Control*, vol. 57, no. 7, pp. 1670–1784, 2012.

[32] I. Pressman and D. Singmaster, "'The Jealous Husbands' and 'The Missionaries and Cannibals'," *The Mathematical Gazette*, vol. 73, no. 464, pp. 73–81, 1989.

[33] S. Reveliotis and A. Nazeem, "Optimal linear separation of the safe and unsafe subspaces of sequential resource allocation systems as a set-covering problem: Algorithmic procedures and geometric insights," *SIAM Journal of Control and Optimization*, vol. 51, no. 2, pp. 1707–1726, 2013.

[34] M. Uzam and M. C. Zhou, "An improved iterative synthesis method for liveness enforcing supervisors of flexible manufacturing systems," *International Journal of Production Research*, vol. 44, no. 10, pp. 1987–2030, 2006.

[35] M. Uzam and M. C. Zhou, "An iterative synthesis approach to Petri net based deadlock prevention policy for flexible manufacturing systems," *IEEE Transactions on Systems, Man and Cybernetics, Part A*, vol. 37, no. 3, pp. 362-371, 2007.

[36] Wikipedia, Missionaries and Cannibals Problem,
http://en.wikipedia.org/wiki/Missionaries_and_cannibals_problem.

[37] N. Q. Wu, M. Zhou, and Z. W. Li, "Resource-oriented Petri net for deadlock avoidance in flexible assembly systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol.38, no.1, pp.56–69, 2008.

[38] K. Yamalidou, J. Moody, M. Lemmon, and P. Antsaklis, "Feedback control of Petri nets based on place invariants," *Automatica*, vol. 32, no. 1, pp. 15–28, 1996.

[39] J. F. Zhang, M. Khalgui, Z. W. Li, G. Frey, O. Mosbahi, H. B. Salah, "Reconfigurable coordination of distributed discrete event control systems", *IEEE Transactions on Control Systems Technology*, in press, DOI: 10.1109/TCST.2014.2313352, 2014.