Petri Net Controllers for Generalized Mutual Exclusion Constraints with Floor Operators

Ziyue Ma, Zhiwu Li, and Alessandro Giua

December, 2016

Abstract

In this paper a special type of nonlinear marking specifications called *stair generalized mutual exclusion constraints* (stair-GMECs) is defined. A stair-GMEC can be represented by an inequality whose left-hand is a linear combination of floor functions. Stair-GMECs have a higher modeling power than classical GMECs and can model legal marking sets that cannot be defined by OR-AND GMECs. We propose two algorithms to enforce a stair-GMEC as a closed-loop net, in which the control structure is composed by a residue counter, remainder counters, and duplicate transitions. We also show that the proposed control structure is maximally permissive since it prevents all and only the illegal trajectories of a plant net. This approach can be applied to both bounded and unbounded nets. Several examples are proposed to illustrate the approach.

Published as:

[Z.Y. Ma, Z.W. Li, A. Giua, "Petri net controllers for Generalized Mutual Exclusion Constraints with floor operators," *Automatica*, Vol. 74, pp. 238-246, **2016**]

DOI: 10.1016/j.automatica.2016.07.042.

Ziyue Ma is with School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China, and also with Dipartimento di Ingegneria Elettrica ed Elettronica, Università degli Studi di Cagliari, Cagliari, Italy (email: maziyue@gmail.com).

Zhiwu Li is with Institute of Systems Engineering, Macau University of Science and Technology, Taipa, Macau (email: zhwli@xidian.edu.cn, systemscontrol@gmail.com).

Alessandro Giua is with Aix Marseille Université, CNRS, ENSAM, Université de Toulon, LSIS UMR 7296, Marseille 13397, France (email: alessandro.giua@lsis.org) and DIEE, University of Cagliari, Cagliari 09124, Italy (email: giua@diee.unica.it).

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61374068, 61472295, the Recruitment Program of Global Experts, the ShaanXi Huashan Scholars, and the Science Technology Development Fund, MSAR, under Grant No. 078/2015/A3.

1 Introduction

Generalized Mutual Exclusion Constraints [9] (GMECs) are a class of state specifications in Petri nets. They can be efficiently enforced on nets if all transitions are controllable by a simple control structure called *monitor places*. Since the monitor design does not require to enumerate the reachability set but is only based on the net structure, the state explosion problem could be avoided and the controller design process is quite efficient. GMECs are an example of Petri net structural approaches that have been proved to be useful in a wide range of contexts [25, 1, 31, 30, 2, 26, 4, 27, 22, 24].

A single GMEC considers a very special class of *legal markings* that satisfy a linear inequality and thus belong to an n-dimensional half-space, where n is the number of places in a net. The legal marking set defined by a set of GMECs is given by the intersection of half-spaces and thus is always convex, as proved in [9]. Single GMECs and conjunction of single GMECs can be enforced in a straightforward manner by adding monitor places [8, 11]. Such approaches have been used in the context of marking specification enforcement problems [25, 10, 11, 12, 13, 2, 3] for *supervisory control* [29] as well as the deadlock prevention in automated manufacturing systems [15, 5, 16, 14]. These results have been generalized showing that a legal marking set that is a finite union of integer convex sets may be defined by a disjunction of GMECs, called an OR-AND GMEC [22], which can be enforced by control structures containing both places and transitions.

Although control structures based on OR-AND GMECs have polynomial complexity with respect to the number of disjunctions in them [22], it is usually not immediate to find an OR-AND GMEC that defines the legal marking set for a given physical model (we discuss this issue in Section 3). In bounded nets, one may need a full enumeration of the reachability space, and it is still difficult and exhaustive to recognize an OR-AND GMEC from a long list of legal markings. In some cases the solution can be obtained stepwise by *GMEC transformations* [20, 18, 19, 21, 32, 23]. However, this type of approaches is only applicable to some very restricted subclasses of Petri nets. Moreover, in some problems the legal marking set is not a finite union of integer convex sets [23], i.e., it cannot be defined by an OR-AND GMEC (examples are given in Sections 3 and 6).

It is worth to note that several alternative methods are also developed to design supervisors based on GMECs. For example, for nets and GMECs that satisfy some structural assumptions, Luo *et al.* developed an efficient supervisor synthesis method that simultaneously performs the reduction of the net structure and the constraint transformation [17]. For a bounded net, a minimal number of disjunctive GMECs that ensures deadlock-free-ness can be obtained by the *set classification* method by Chen *et al.* [6]. However, it can only be applied to small-scale systems, and the set of forbidden markings must be convex and must be known *in prior*. Moreover, Qin *et al.* developed a method to obtain a controller that ensures liveness in LS^3PR nets, which circumvents the GMEC transformation [28]. However, the controller by this method is in general not maximally permissive, i.e., some legal markings may not be reached in the closed-loop net. In this sense, the classical GMEC approach needs to be further extended, and new types of constraints with a higher modeling power while still enforceable with a simple control structure are required.

To overcome the aforementioned difficulties, this paper proposes a new type of nonlinear marking specifications called *stair-GMECs* (defined in Section 3). In particular, they are extremely suitable for characterizing the set of admissible markings in many supervisory control problems for Petri nets containing uncontrollable transitions. Such sets are often very complex and cannot always be efficiently — if at all — described by OR-AND GMECs [23].

A class of nonlinear marking constraints in Petri nets has also been studied by Chen *et al.* in [7] where a method was proposed to enforce an *additive separable nonlinear constraint* (ASNC). Their approach works in a straightforward manner, but is only applicable for Petri nets with a known bound. Moreover, the structural complexity of the resulting controller is not satisfactory.

The modeling power of stair-GMECs and ASNCs are not comparable, and hence we present a different approach to enforce a stair-GMEC in this paper. We also note that for a problem that can be modeled by both types of constraints, the control structure that we propose is in general more compact than that of [7]. The contributions of this paper are summarized as follows:

- A new type of nonlinear marking specifications for Petri nets, i.e., stair-GMECs, is proposed. A stair-GMEC can be represented as an integer inequality whose left-hand side is a linear combination of certain *floor functions* $\lfloor \cdot \rfloor$. We show that stair-GMECs can conveniently characterize legal marking sets that are difficult or even not possible to define by OR-AND GMECs. We also prove that under certain restrictive conditions a stair-GMEC can be converted to an equivalent GMEC.
- Two algorithms are developed to design a controller that enforces a given stair-GMEC. The control structure consists of two parts: (1) newly added control places including one *residue counter* and a series of *remainder counters*, and (2) transitions duplicated from plant transitions. We prove that this control structure is maximally permissive, i.e., it prevents all and only the illegal trajectories of the plant.
- The control structure has a relatively compact structure comparing with those in [22] and in [7], and a detailed comparison including an example is given at the end of this paper. Furthermore, this approach can be applied to both bounded and unbounded nets.

The paper is organized in seven sections. Section 2 recalls the Petri net formalism used in the paper. Section 3 introduces stair-GMECs and some properties are studied. Section 4 develops an algorithm to construct the Petri net controller to enforce a stair-GMEC, and its maximal permissiveness is proved. The complexity analysis of this approach is given in Section 5. An illustrative example is presented in Section 6, and conclusions are drawn in Section 7.

2 Preliminaries

2.1 Petri net

A Petri net is a four-tuple $N = (P, T, Pre, Post)$, where P is a set of m *places* graphically represented by circles; T is a set of n *transitions* graphically represented by boxes; $Pre : P \times T \to \mathbb{N}$ and $Post : P \times T \to \mathbb{N}$ are the *pre-* and *post-incidence functions* that specify the arcs in the net and are represented as matrices in $\mathbb{N}^{m\times n}$ ($\mathbb{N} = \{0, 1, 2, \ldots\}$). The *incidence matrix* of a net is defined by $C = Post - Pre \in \mathbb{Z}^{m\times n}$ ($\mathbb{Z} = \{0, \pm 1, \pm 2, \ldots\}$). A net is said to be *self-loop free* if $\forall (p,t) \in P \times T$, $Pre(p,t) \cdot Post(p,t) = 0$ holds. For a self-loop free net, from the incidence matrix one may univocally determine the *Pre* and *Post* functions.

For a transition $t \in T$ we define *the set of its input places* as ${}^{\bullet}t = \{p \in P \mid Pre(p,t) > 0\}$ and *the set of its output places* as $t^{\bullet} = \{p \in P \mid Post(p,t) > 0\}$.

A *marking* is a vector $M : P \to \mathbb{N}$ that assigns to each place of a Petri net a non-negative integer number of tokens, graphically represented by black dots. We denote by $M(p)$ the marking of place p. A *marked net*, also called a *plant*, $\langle N, M_0\rangle$ is a net N with an initial marking M_0.

A transition t is *enabled* at M if $M \geq Pre(\cdot,t)$ and may fire reaching a new marking M' with $M' = M + C(\cdot,t)$. We write $M[\sigma\rangle$ to denote that the sequence of transitions $\sigma = t_{j1} \cdots t_{jk}$ is enabled at M, and we write $M[\sigma\rangle M'$ to denote that the firing of σ at M yields M'.

A marking M is *reachable* in $\langle N, M_0\rangle$ if there exists a firing sequence σ such that $M_0[\sigma\rangle M$. The set of all markings reachable from M_0 defines the *reachability set* of $\langle N, M_0\rangle$ and is denoted by $R(N, M_0)$. The set of all firable sequences from M_0 defines the *language* of $\langle N, M_0\rangle$ and is denoted by $L(N, M_0)$. A place $p \in P$ of a marked net $\langle N, M_0\rangle$ is said to be *bounded* if there exists a nonnegative integer K such that for all marking $M \in R(N, M_0)$, $M(p) \leq K$ holds, and the minimal value of K is said to be the *bound* of place p. A marked net $\langle N, M_0\rangle$ is bounded if all its places are bounded.

We use $\lfloor \cdot \rfloor$ to denote the maximal integer that does not exceed $(\cdot)$.

2.2 GMEC

Definition 1 *A* Generalized Mutual Exclusion Constraint *(GMEC) is a pair* $(\mathbf{w},k)$ *that defines a set of legal markings:*

$$\mathscr{L}_{(\mathbf{w},k)} = \{M \in \mathbb{N}^m \mid \mathbf{w}^T \cdot M \leq k\}$$

where $\mathbf{w} \in \mathbb{Z}^m$ *and* $k \in \mathbb{Z}$. △

Definition 2 *An AND-GMEC is a set of single GMECs denoted by a pair* $(\mathbf{W},\mathbf{k})$ *where* $\mathbf{W} = [\mathbf{w}_1 \cdots \mathbf{w}_s] \in \mathbb{Z}^{m\times s}$ *and* $\mathbf{k} = [k_1 \cdots k_s]^T \in \mathbb{N}^s$*. An AND-GMEC defines a set of legal markings*

$$\mathscr{L}_{(\mathbf{W},\mathbf{k})} = \{M \in \mathbb{N}^m \mid \forall(\mathbf{w}_i,k_i) \in (\mathbf{W},\mathbf{k}), \mathbf{w}_i^T \cdot M \leq k_i\}.$$

An OR-AND GMEC is a set $W = \{(\mathbf{W}_1,\mathbf{k}_1), \ldots, (\mathbf{W}_r,\mathbf{k}_r)\}$ *in which each* $(\mathbf{W}_i,\mathbf{k}_i) \in \mathbb{Z}^{m\times s_i} \times \mathbb{N}^{s_i}$ *is an AND-GMEC for* $1 \leq i \leq r$*, and r is called the* number of disjunctions *of* W*. An OR-AND GMEC defines a set of legal markings:*

$$\mathscr{L}_W = \{M \in \mathbb{N}^m \mid \exists(\mathbf{W}_i,\mathbf{k}_i) \in W, \mathbf{W}_i^T \cdot M \leq \mathbf{k}_i\}.$$

△

A single GMEC $(\mathbf{w},k)$ on a plant $\langle N,M_0\rangle$ with $N = (P,T,Pre,Post)$ can be enforced through a control structure by adding to the net a loop-free place q called *the monitor place* which has an incidence matrix row $C(q,\cdot) = -\mathbf{w}^T \cdot C(\cdot,t)$ and is initially marked as $M_0(q) = k - \mathbf{w}^T \cdot M_0$ [9].

3 Stair-GMECs and the Problem Formulation

3.1 GMECs and Stair-GMECs

In many supervisory control problems in Petri nets, the legal marking set can be written as a finite union of integer convex sets[1] In such a case the legal marking set can always be written as an OR-AND GMEC W and can then be enforced by a place/transition controller that is maximally permissive if W is bounded [22].

Although control structures based on OR-AND GMECs have polynomial complexity with respect to the number of disjunctions in them [22], it is not always convenient to characterize a legal marking set of a physical model by OR-AND GMECs. For example, there are cases in which each AND-GMEC is used

[1]A set $X \subseteq \mathbb{R}^m$ is *convex* if $(x_1,x_2 \in X) \Rightarrow (\forall\lambda \in [0,1], \lambda\cdot x_1 + (1-\lambda)\cdot x_2 \in X)$. A set $S \subseteq \mathbb{N}^m$ is said to be an *integer convex set* if there exists a convex set $X \subseteq \mathbb{R}^m$ such that $S = X \cap \mathbb{N}^m$.

to describes just a small number of legal markings and thus the OR-AND GMEC contains a number of disjunctive terms of order comparable to the cardinality of the legal marking set as shown in the following Example 1. The same example shows that stair-GMECs can describe some particular classes of infinite legal marking sets that are not a finite union of convex sets, and thus cannot be described by OR-AND GMECs as discussed in [23]. As a result, in this section we propose a new type of constraints called *stair-GMECs* that have a higher modeling power than classical GMECs but are still enforceable with a simple control structure.

Definition 3 *A* stair-GMEC *is a four-tuple* $(\mathbf{A},\mathbf{b},\mathbf{c},k)$*, where* $\mathbf{A} = [\mathbf{a}_1 \cdots \mathbf{a}_s] \in \mathbb{Z}^{m\times s}$*,* $\mathbf{b} \in \mathbb{Z}^s$*,* $\mathbf{c} \in \mathbb{N}^s$*, and* $k \in \mathbb{N}$*. A stair-GMEC* $(\mathbf{A},\mathbf{b},\mathbf{c},k)$ *defines a set of legal markings* $\mathscr{L}_{(\mathbf{A},\mathbf{b},\mathbf{c},k)}$*:*

$$\mathscr{L}_{(\mathbf{A},\mathbf{b},\mathbf{c},k)} = \{M \in \mathbb{N}^m \mid \sum_{i=1}^{s} b_i \cdot \left\lfloor \frac{\mathbf{a}_i^T \cdot M}{c_i} \right\rfloor \leq k\}, \tag{1}$$

where $\lfloor x \rfloor$ *denotes the largest integer less than or equal to x.* △

In general, due to the floor operator $\lfloor\cdot\rfloor$ in Eq. (1), a stair-GMEC is not a linear marking specification, and the legal marking set of a stair-GMEC is not convex. However, a single GMEC $(\mathbf{w},k)$ is always a stair-GMEC $(\mathbf{w},1,1,k)$.

For convenience of notation, in the following we also propose an equivalent expression to describe a stair-GMEC, which will be used in the remaining part of this paper. The left-hand side of the inequality in Eq. (1) can be considered as the sum of s functions $f_i(M) = b_i \cdot \lfloor \mathbf{a}_i^T \cdot M/c_i \rfloor$, $i = 1,\ldots,s$. By denoting $\xi(M) = \sum_{i=1}^{s} f_i(M)$, a stair-GMEC $(\mathbf{A},\mathbf{b},\mathbf{c},k)$ can be equivalently defined as a pair (ξ,k) whose legal marking set is:

$$\mathscr{L}_{(\mathbf{A},\mathbf{b},\mathbf{c},k)} = \mathscr{L}_{(\xi,k)} = \{M \in \mathbb{N}^m \mid \xi(M) = \sum_{i=1}^{s} f_i(M) \leq k\}, \tag{2}$$

and in the sequel of this paper we will use (ξ,k) to denote a stair-GMEC. Hence a marking is legal if it satisfies the following inequality:

$$\xi(M) = \sum_{i=1}^{s} f_i(M) = \sum_{i=1}^{s} b_i \left\lfloor \frac{\mathbf{a}_i^T \cdot M}{c_i} \right\rfloor \leq k. \tag{3}$$

Remark 1 *A stair-GMEC* (ξ,k) *reduces to a GMEC if* $s = 1$ *and* $b_1 = c_1 = 1$ *in Eq.*(3). △

Stair-GMECs are a generalization of GMECs that allow to compactly describe meaningful classes of legal marking sets that are not convex. Thus they allow to solve control problems for which past methods are either too complex in implementation (such is the case of bounded nets using OR-AND GMECs or other non-linear methods such as [7]) or cannot apply (such is the case of unbounded nets). This is illustrated by the following example. A comprehensive example is also given in Section 6.

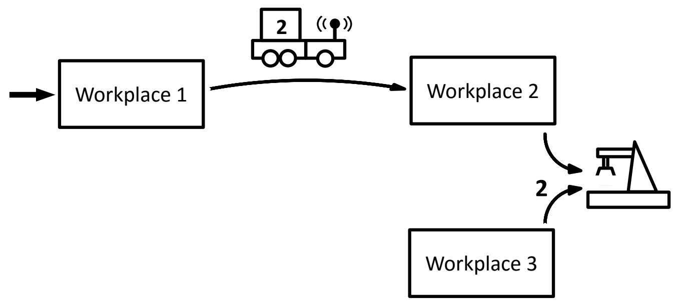

Figure 1: A manufacturing system.

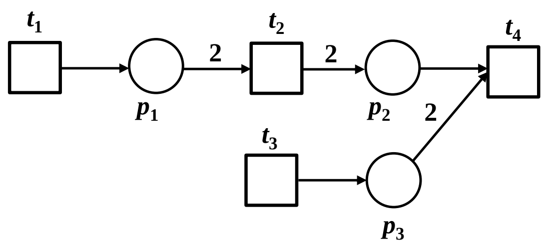

Figure 2: The Petri net model of the system in Figure 1.

Example 1 *Consider the manufacturing system in Figure 1 whose corresponding Petri net model is shown in Figure 2. Raw parts of type A arrive (t_1) in Workplace 1 (p_1) and then are loaded on an AGV (t_2) to be transferred to Workplace 2 (p_2). The AGV is set to transport two parts of type A from Workplace 1 to Workplace 2 automatically once it is fully loaded. A robot (t_4) takes one part of type A from Workplace 2 and two raw parts of type B arriving (t_3) in Workplace 3 (p_3) to produce the final product.*

We consider a safety constraint related to the number of parts of type A. If the arrival of raw parts of type B is unexpectedly delayed, the blocked parts of type A in Workplace 2 (i.e., after consuming all parts of type B in Workplace 3) must not exceed 5. Noting that the AGV will automatically transfer parts of type A from Workplace 1 to Workplace 2, the legal marking set $\mathscr{L}$ to be enforced can be defined by the following stair-GMEC (ξ, k) solely:

$$2\lfloor M(p_1)/2 \rfloor + M(p_2) - \lfloor M(p_3)/2 \rfloor \leq 5 \tag{4}$$

A similar example is also discussed in [23] where it is shown that the legal set for this problem, i.e., $\mathscr{L}_{(\xi,k)}$, cannot be defined by an OR-AND GMEC.

On the other hand, let us assume that p_3 has a bound $K \in \mathbb{N}$. In such a case the legal marking set $\mathscr{L}$ is *characterized by the following inequalities:*

$$\begin{cases} 2\lfloor M(p_1)/2 \rfloor + M(p_2) - \lfloor M(p_3)/2 \rfloor & \leq 5 \\ M(p_3) & \leq K \end{cases} \tag{5}$$

The set $\mathscr{L}$ is now finite, and hence there exists an OR-AND GMEC defining it. However, one can readily verify that it is difficult to obtain such an OR-AND GMEC.[2] *In fact, $\mathscr{L}$ can be defined by the following OR-AND GMEC W:*

$$\bigvee_{j=0}^{\lfloor K/4+5/2 \rfloor} (M(p_1) \leq 2j+1) \wedge (2M(p_2) - M(p_3) \leq 10 - 4j) \tag{6}$$

This OR-AND GMEC consists of $\lfloor K/4+5/2 \rfloor + 1$ AND-GMECs and it cannot be simplified anymore. Obviously, it is much simpler to define $\mathscr{L}$ by Eq. (5) *instead of Eq.* (6). $\triangle$

Remark 2 *We also point out that to enforce the constraint in Eq.* (6) *by the method proposed in [22],* $3 \times \lfloor K/4+5/2 \rfloor$ *additional places (called* control places*) have to be added, and* $(\lfloor K/4+5/2 \rfloor)^2$ *additional transitions (called* mirror transitions*) have to be added for each* migrating transition *[22]. For example, for $K = 120$ there will be* 96 *control places and* 1,024 *duplicate transitions for transition t_2, and a polling mechanism has to be introduced to circularly activate them. Moreover, once the value of K changes, the corresponding controller has to be completely redesigned. On the contrary, it is much simpler to define the legal marking set by a stair-GMEC, e.g., Eq.* (4). *Furthermore, in the sequel we show that a stair-GMEC can be easily enforced as a closed-loop net.* $\triangle$

In general a stair-GMEC defines a set of legal markings that do not form a half-space, and therefore it cannot be enforced by monitor places as in [9]. However, in some cases a stair-GMEC may define a half-space and hence is equivalent to a single GMEC.

Proposition 1 *The stair-GMEC* (ξ, k) *where*

$$\xi(M) = \left\lfloor \frac{\mathbf{a}_1^T \cdot M}{c} \right\rfloor + \left\lfloor \frac{\mathbf{a}_2^T \cdot M}{1} \right\rfloor \tag{7}$$

is equivalent to the single GMEC $(\mathbf{w}', k')$ *(i.e.,* $\mathscr{L}_{(\xi,k)} = \mathscr{L}_{(\mathbf{w}',k')}$*) where*

$$\begin{cases} \mathbf{w}' = \mathbf{a}_1^T + c \cdot \mathbf{a}_2^T \\ k' = (k+1) \cdot c - 1. \end{cases} \tag{8}$$

[2] The OR-AND GMEC that defines a given finite marking set $\mathscr{L}$ is in general not unique. However, there is no efficient method to obtain it. For example, a brute-force way is to characterize a finite $\mathscr{L}$ by: $\bigvee_{M' \in \mathscr{L}} \bigwedge_{i=1}^{m} (M(p_i) \leq M'(p_i)) \wedge (-M(p_i) \leq -M'(p_i))$ which is an OR-AND GMEC. However, the resulting OR-AND GMEC consists of $O(2m \cdot |\mathscr{L}|)$ single GMECs. Such complexity is too high to be feasible in practice.

Proof: First, it is not difficult to prove that given $x, y \in \mathbb{Z}$ and $z \in \mathbb{N}$, it holds:

$$\lfloor x/z \rfloor \leq y \qquad \Leftrightarrow \qquad x \leq (y+1) \cdot z - 1 \tag{9}$$

Then we have

$$\begin{aligned} M \in \mathscr{L}_{(\xi,k)} &\Leftrightarrow \left\lfloor \frac{\mathbf{a}_1^T \cdot M}{c} \right\rfloor + \left\lfloor \frac{\mathbf{a}_2^T \cdot M}{1} \right\rfloor \leq k \\ &\Leftrightarrow \left\lfloor \frac{\mathbf{a}_1^T \cdot M}{c} \right\rfloor \leq k - \mathbf{a}_2^T \cdot M \\ &\overset{(9)}{\Longleftrightarrow} \mathbf{a}_1^T \cdot M \leq ((k - \mathbf{a}_2^T \cdot M) + 1) \cdot c - 1 \\ &\Leftrightarrow (\mathbf{a}_1^T + c \cdot \mathbf{a}_2^T) \cdot M \leq (k+1) \cdot c - 1 \\ &\Leftrightarrow M \in \mathscr{L}_{(\mathbf{w}',k')} \end{aligned}$$

□ ■

Note that the type of stair-GMEC considered in Proposition 1 is rather peculiar: it contains only one term containing the floor operator with $b_1 = 1$. In particular, given a stair-GMEC (ξ, k), if in $\xi(M)$ there exists one term $b_i \cdot \lfloor \mathbf{a}_i^T \cdot M/c_i \rfloor$ where $b_i, c_i > 1$, or there exists two terms $\lfloor \mathbf{a}_1^T \cdot M/c_1 \rfloor + \lfloor \mathbf{a}_2^T \cdot M/c_2 \rfloor$ where $c_1, c_2 > 1$, then $\mathscr{L}_{(\xi,k)}$ would not be a half-space and there does not exist a single GMEC equivalent to it.[3]

Example 2 *Consider the stair-GMEC represented by the following inequality:*

$$\left\lfloor \frac{M(p_1) - 2M(p_2) + M(p_3)}{3} \right\rfloor + 2M(p_1) - M(p_3) \leq 3.$$

It can be rewritten as:

$$\left\lfloor \frac{M(p_1) - 2M(p_2) + M(p_3)}{3} \right\rfloor + \left\lfloor \frac{2M(p_1) - M(p_3)}{1} \right\rfloor \leq 3.$$

Then by Proposition 1 it can be converted to an equivalent GMEC represented by the following inequality:

$$7M(p_1) - 2M(p_2) - 2M(p_3) \leq 11.$$

△

At the end of this subsection we point out that, since a single GMEC is also a stair-GMEC, for any OR-AND GMEC there exists a conjunction/disjunction of stair-GMECs defining the same legal marking set. Due to the limit of space, we do not study conjunction/disjunction of stair-GMECs in this paper but will explore it in the future. Note that the converse does not hold: as we have shown in Example 4, there exists some

[3] In this discussion we assume that k is sufficient large to ensure that $\mathscr{L}_{(\xi,k)}$ is not reduced to some very simple integer sets, e.g., a singleton or the emptyset.

stair-GMEC whose legal marking set cannot be defined by an OR-AND GMEC.

3.2 Problem Formulation

In the classical AND-GMEC controller design approaches, a controller consists of a set of monitor places P_S that are added to a plant net to determine a closed-loop net $\hat{N}$ with $\hat{P} = P \cup P_S$ and $\hat{T} = T$. However, for stair-GMECs (as well as other nonlinear constraints) it is not in general possible to build a controller that only consists of control places. Hence we look for a control structure that contains both additional control places and control transitions.

We will consider a transition set $\hat{T}$ in the closed-loop net that only contains transitions that are duplicate of transitions in the open-loop plant. A transition $\hat{t}$ is called a *duplicate* of a plant transition $t \in T$ if $\hat{t}$ has the same input and output plant places as that of t, and the corresponding weight of arcs are identical [8]. In the next section we propose a control structure that consists of control places and only duplicate transitions. The problem studied in this paper is stated after defining duplicate transitions.

Definition 4 *Given a Petri net $\langle N, M_0 \rangle$ and a closed-loop net $\langle \hat{N}, \hat{M}_0 \rangle$ where $N = (P, T, Pre, Post)$ and $\hat{N} = (P \cup P_S, \hat{T}, \hat{P}re, \hat{P}ost)$, a transition $\hat{t} \in \hat{T}$ is a* duplicate *of a plant transition $t \in T$ if $\forall p \in {}^{\bullet}\hat{t} \cap P, Pre(p, \hat{t}) = Pre(p, t)$ and $\forall p \in \hat{t}^{\bullet} \cap P, Post(p, \hat{t}) = Post(p, t)$. The set of duplicate transitions of $t \in T$ is denoted as $\mathscr{T}(t)$.*
$\triangle$

Problem 1 *Given a net $\langle N, M_0 \rangle$ where $N = (P, T, Pre, Post)$ and a stair-GMEC (ξ, k), determine a closed-loop net $\langle \hat{N}, \hat{M}_0 \rangle$ with $\hat{N} = (P \cup P_S, \hat{T}, \hat{P}re, \hat{P}ost)$ such that $\hat{T} = \bigcup_{t \in T} \mathscr{T}(t)$ and the projection of the reachability set of the net $\hat{N}$ on the set of places P of N satisfies $R(\hat{N}, \hat{M}_0)_{\uparrow P} \subseteq \mathscr{L}_{(\xi,k)}$. To ensure the existence of a solution, we assume that the initial marking is legal, i.e., $M_0 \in \mathscr{L}$.* $\triangle$

4 Controller Design for Stair-GMECs

4.1 Controller Design

We first propose some definitions that will be used in the algorithms.

Definition 5 *Given a net $\langle N, M_0 \rangle$ and a function $f(M) = b \cdot \lfloor \mathbf{a}^T \cdot M / c \rfloor$, the* influence *of a transition $t \in T$ on f is defined as: $\eta(f, t) = \mathbf{a}^T \cdot C(\cdot, t)$.* $\triangle$

Definition 6 *Given a stair-GMEC* (ξ, k) *with* $\xi(M) = \sum_{i=1}^{s} f_i(M)$, $f_i(M) = b_i \cdot \lfloor \mathbf{a}_i^T \cdot M/c_i \rfloor$, *the quantity of* $k - \xi(M) = k - \sum_{i=1}^{s} f_i(M)$ *is called the* available residue *of a marking* M. △

Algorithm 1 given below determines a control structure that is capable of enforcing a given stair-GMEC. By Algorithm 1 a *residue place* is added, and for each $f_i(M) = b_i \cdot \lfloor \mathbf{a}_i^T \cdot M/c_i \rfloor$, c_i *remainder places* are added. The use of the residue place is to record the difference between $\xi(M)$ and the constraint bound k. On the other hand, at any marking M for each $f_i(M)$ there exists a unique remainder place being marked, corresponding to the reminder of $\mathbf{a}_i \cdot M/c_i$. During its execution, Algorithm1 calls Algorithm 2 which generates the duplicate of transitions.

Algorithm 1 Controller Design for a Stair-GMEC

Input: A Petri net $\langle N, M_0 \rangle$ and a stair-GMEC (ξ, k) where $N = (P, T, Pre, Post)$, $\xi(M) = \sum_{i=1}^{s} f_i(M)$, and $f_i(M) = b_i \cdot \lfloor \mathbf{a}_i^T \cdot M/c_i \rfloor$;
Output: A closed-loop net $\langle \hat{N}, \hat{M}_0 \rangle$;

1: Let $\hat{N} = N$, let $\hat{P} = P \cup \{p_s\}$;
2: Let $\hat{M}_0(p_s) = k - \sum_{i=1}^{s} b_i \cdot \lfloor \mathbf{a}_i^T \cdot M_0/c_i \rfloor$;
3: **for all** f_i, **do**
4:     Let $\hat{P} = P \cup \{q_i^0, \ldots, q_i^{c_i-1}\}$;
5:     Let $r = \mathbf{a}_i^T \cdot M_0 \mod c_i$;
6:     Let $\hat{M}_0(q_i^r) = 1, \hat{M}_0(q_i^j) = 0$ for $j \neq r$.
7:     **for all** $t_x \in T$, **do**
8:         Call Algorithm 2 to duplicate t_x;
9:     **end for**
10: **end for**
11: Let $\hat{M}_0(p) = M_0(p)$ for all $p \in P$;
12: Remove isolated places in $\hat{P}$;
13: Output $\langle \hat{N}, \hat{M}_0 \rangle$ where $\hat{N} = (\hat{P}, \hat{T}, \hat{P}re, \hat{P}ost)$.

Algorithm 2 Duplication of Transitions

Input: A Petri net $\langle \hat{N}, \hat{M}_0 \rangle$, a function $f_i(M) = b_i \cdot \lfloor \mathbf{a}_i^T \cdot M/c_i \rfloor$, and a transition $t_x \in T$;
Output: An updated $\langle \hat{N}, \hat{M}_0 \rangle$;

1: **if** $\eta(f_i, t_x) \mod c_i \equiv 0$, **then**
2:     Rename t_x as t_x^0;
3:     Let $\hat{C}(p_s, t_x) = \hat{C}(p_s, t_x) - b_i \cdot \lfloor \eta(f_i, t_x)/c_i \rfloor$;
4: **else**
5:     Let $\hat{T} = \hat{T} \cup \{t_x^0, \ldots, t_x^{c_i-1}\}$, let $\hat{T} = \hat{T} \setminus \{t_x\}$;
6:     **for all** $j = 0$ to $c_i - 1$, **do**
7:         Let $Pre(\cdot, t_x^j) = Pre(\cdot, t_x)$, $Post(\cdot, t_x^j) = Post(\cdot, t_x)$;
8:         Let $Pre(q_i^j, t_x^j) = Post(q_i^y, t_x^j) = 1$ where $y = [j + \eta(f_i, t_x)] \mod c_i$;
9:         Let $\hat{C}(p_s, t_x^j) = \hat{C}(p_s, t_x^j) - b_i \cdot \lfloor (j + \eta(f_i, t_x))/c_i \rfloor$;
10:     **end for**
11: **end if**
12: Return $\langle \hat{N}, \hat{M}_0 \rangle$;

We briefly explain how Algorithms 1 and 2 work before presenting an example. In Steps 1 and 2 of Algorithm 1, a *residue place* p_s is added and initially marked by $k - \xi(M_0)$ tokens, i.e., the available residue of M_0. Then by the loop from Steps 3 to 10 each f_i is treated sequentially. In the first iteration, in Step 4 a set

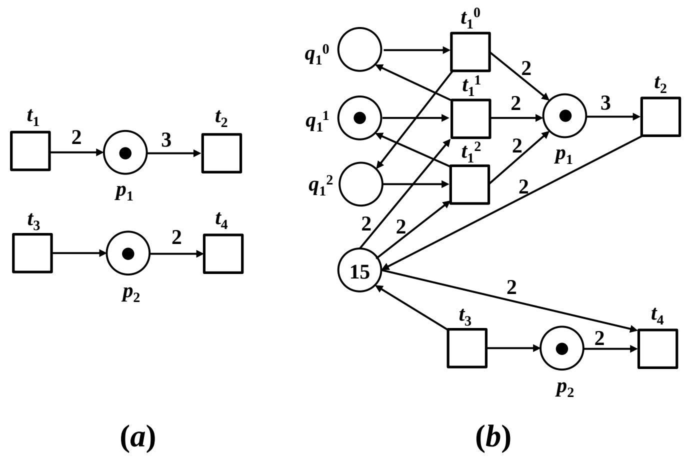

Figure 3: An illustration of Algorithms 1 and 2.

of *remainder places* $q_1^0, \ldots, q_1^{c_1-1}$ are added. In Steps 5 and 6 the r-th place in these remainder places (i.e., q_1^r) is marked with one token called the *remainder token*, where r is the remainder of $\mathbf{a}_1^T \cdot M_0/c_1$. In fact, during the evolution of the closed-loop net, at an arbitrary marking M this remainder token would mark the place q_1^j among these remainder places of f_1 where j is the remainder of $\mathbf{a}_1^T \cdot M/c_1$. Then by Step 8 Algorithm 1 calls Algorithm 2 to duplicate each transition $t_x \in T$.

In Algorithm 2, if the influence of t_x on f_1 is a multiple of c_1, i.e., $\eta(f_1, t_x) \mod c_1 = 0$, then t_x is not duplicated (since its firing will not change the remainder of $\mathbf{a}_1^T \cdot M/c_1$) but is renamed for convenience of notation. The change of $f_i(M)$ by firing t_x is recorded into the arc between p_s and t_x. On the other hand, if $\eta(f_1, t_x)$ is not a multiple of c_1, then t_x is duplicated to c_1 transitions $t_x^0, \ldots, t_x^{c_1-1}$ by Step 5. By Steps 7 and 8, the firing of the duplicate transition t_x^j of t_x would move the remainder token from place q_1^j to q_1^y to update the current remainder of $\mathbf{a}_i^T \cdot M/c_i$. By Step 9, the change of $f_i(M)$ by firing t_x in case that the remainder of $\mathbf{a}_i^T \cdot M/c_i$ is j is recorded into the arc between p_s and t_x. After all transitions are treated, it returns to Algorithm 1. Then Algorithm 1 goes back to Step 3 to treat f_2. This process continues until all f_i's are processed. We illustrate Algorithms 1 and 2 by the following example.

Example 3 *Consider the net $\langle N, M_0 \rangle$ in Figure 3(a) (where $M_0 = [1,1]^T$) and a stair-GMEC (ξ, k) to be enforced: $\xi(M) = f_1(M) + f_2(M) = 2\lfloor M(p_1)/3 \rfloor - \lfloor M(p_2)/1 \rfloor \leq 14$. In the beginning of Algorithm 1, a control place p_s is added and initially marked with $k - \xi(M_0) = 15$ tokens.*

In the first loop, f_1 is treated and $c_1 = 3$ remainder places q_1^0, q_1^1, and q_1^2 are added to $\hat{P}$. The remainder place q_1^1 is marked with one token since $\mathbf{a}_1^T \cdot M_0/c_1 = 1$. By calling Algorithm 2, since $\eta(f_1, t_1) \mod c_1 \equiv 2 \neq 0$, three duplicate transitions of t_1 are added and the corresponding arcs are calculated. For example, if the current remainder of $\mathbf{a}_1^T \cdot M/c_1 = 1$ is 1, only t_1^1 is enabled. The firing of t_1^1 will move the remainder token from q_1^1 to q_1^0 since by firing t_1 the new remainder of $\mathbf{a}_1^T \cdot M/c_1 = 1$ is 0, while $\hat{C}(p_s, t_1^1) = -2$ since the firing *of t_1 at a marking M such that $\mathbf{a}_1^T \cdot M \mod c_1 \equiv 1$ will increase the value of $f_1(M)$ by 2. For transitions t_2, t_3, and t_4, since $\eta(f_1,t_2) \mod c_1 = \eta(f_1,t_3) \mod c_1 = \eta(f_1,t_4) \mod c_1 \equiv 0$, no duplicate transitions are added for t_2, t_3, nor t_4. Then the first loop of Algorithm 1 is done. In the second loop, since $c_2 = 1$, no more duplicate transitions are added. Finally we obtain the closed-loop net $\langle \hat{N}, \hat{M}_0 \rangle$ in Figure 3(b).* $\triangle$

4.2 Correctness and Maximal Permissiveness of Algorithm 1

In this subsection we first prove that Algorithm 1 is correct. The following proposition shows that given a firing trajectory in the plant net $\langle N, M_0 \rangle$, there exists a unique firing trajectory in the closed-loop net $\langle \hat{N}, \hat{M}_0 \rangle$ composed of corresponding duplicate transitions.

Proposition 2 *Given a Petri net $\langle N, M_0 \rangle$ and a stair-GMEC (ξ, k) with $\xi(M) = \sum_{i=1}^{s} f_i(M)$, $f_i(M) = b_i \cdot \lfloor \mathbf{a}_i^T \cdot M/c_i \rfloor$, let $\langle \hat{N}, \hat{M}_0 \rangle$ be the closed-loop net obtained by Algorithm 1. If $M_0[t\rangle_N M_1$ satisfying $\xi(M_1) \leq k$, then in $\hat{N}$ there exists a unique $\hat{t} \in \mathscr{T}(t)$ such that $\hat{M}_0[\hat{t}\rangle_{\hat{N}} \hat{M}_1$ and:*

1. *$\hat{M}_{1\uparrow P} = M_1$;*
2. *for all $i \in \{1, \ldots, s\}$, $\hat{M}_1(q_i^{y_i}) = 1$ where $\mathbf{a}_i^T \cdot M_1 \mod c_i \equiv y_i$, and $\hat{M}_1(q_i^z) = 0$ if $z \neq y_i$;*
3. *$\hat{M}_1(p_s) = k - \xi(M_1)$.*

Proof: We first prove that given an arbitrary transition $t_x \in T$ such that $M_0[t\rangle_N M_1$ and $\xi(M_1) \leq k$, there exists a unique duplicate transition of t_x that can fire at $\hat{M}_0$ in $\hat{N}$.

Let $Q_i = \{q_i^0, \ldots, q_i^{c_i-1}\} \subseteq P_S$ denote the set of remainder places added for f_i. Consider t_x and the first function f_1 in ξ. If $\eta(f_1,t_x) \mod c_1 \equiv 0$, then the renamed transition t_x^0 is not disabled by any places in Q_1. If $\eta(f_1,t_x) \mod c_1 \neq 0$, then t_x is duplicated to c_1 transitions by Algorithm 2 among which there is a unique transition $t_x^{r_1}$ that is not disabled by Q_1, since ${}^\bullet t_x^{r_1} \cap Q_1 = \{q_1^{r_1}\}$ where r_1 is the remainder of $\mathbf{a}_1^T \cdot M/c_1$ and $q_1^{r_1}$ is marked at M_0. The same reasoning can be applied to $t_x^{r_1}$ and f_2 such that there exists a unique $t_x^{r_1 r_2} \in \mathscr{T}(t_x)$ which is not disabled by neither Q_1 nor Q_2. Hence finally there exists a unique transition $t_x^{r_1 \cdots r_s}$ that is a duplicate transition of t_x, and $t_x^{r_1 \cdots r_s}$ is not disabled by Q_i for all $i = 1, \ldots, s$. By $t_x^{r_1 \cdots r_s} \in \mathscr{T}(t_x)$, $t_x^{r_1 \cdots r_s}$ is not disabled by any places $p \in P$ at $\hat{M}_0$.

Now let us consider the residue place p_s. The change of f_i by firing t_x is $f_i(M_1) - f_i(M_0) = b_i \cdot \lfloor (r_i + \eta(f_i,t_x))/c_i \rfloor$ where r_i is the remainder of $\mathbf{a}_i^T \cdot M_0/c_i$. Hence the change of $\xi(M)$ is $\xi(M_1) - \xi(M_0) = \sum_{i=1}^{s}(f_i(M_1) - f_i(M_0)) = -\hat{C}(p_s, t_x^{r_1 \cdots r_s})$. Since $\hat{M}_0(p_s) = k - \xi(M_0)$, $M_1 \in \mathscr{L}_{(\xi,k)}$, and there is no self-loop between p_s and $t_x^{r_1 \cdots r_s}$, we have $M_0(p_s) - Pre(p_s, t_x^{r_1 \cdots r_s}) = M_0(p_s) - \hat{C}(p_s, t_x^{r_1 \cdots r_s}) = k - \xi(M_0) - (\xi(M_1) - \xi(M_0)) = k - \xi(M_1) \geq 0$. This indicates that transition $t_x^{r_1 \ldots r_s}$ is not disabled by p_s at M_0. As a result, $t_x^{r_1 \ldots r_s}$

is the unique duplicate transition of t_x that can fire at $\hat{M}_0$, leading to a unique marking $\hat{M}_1$ in the closed-loop net $\hat{N}$.

Now we show that the three conditions in the statement holds. Since $t_x^{r_1\cdots r_s} \in \mathscr{T}(t)$, $\hat{M}_{1\uparrow P} = M_1$ holds. By Steps 7 and 8 in Algorithm 2, the firing of $t^{r_1\cdots r_x}$ would move each unique remainder token in Q_i from place $q_i^{r_i}$ to $q_i^{y_i}$ where $\mathbf{a}_i^T \cdot M_1 \mod c_i \equiv y_i$, and hence Condition 2 holds. Finally, $\xi(M_1) = M_0(p_s) - \hat{C}(p_s, t_x^{r_1\cdots r_s}) = k - \xi(M_1)$ holds, which concludes the proof. □ ■

Based on the previous proposition, we can now prove the following main result.

Theorem 1 *Given a Petri net $\langle N, M_0\rangle$ and a stair-GMEC (ξ, k) with $\xi(M) = \sum_{i=1}^{s} f_i(M)$, $f_i(M) = b_i \cdot \lfloor \mathbf{a}_i^T \cdot M/c_i \rfloor$, the closed-loop net $\langle \hat{N}, \hat{M}_0\rangle$ obtained by Algorithm 1 satisfies $R(\hat{N}, \hat{M}_0)_{\uparrow P} \subseteq \mathscr{L}_{(\xi,k)}$.*

Proof: Consider an arbitrary firing trajectory:

$$M_0[t_0\rangle_N M_1 \cdots M_{x-1}[t_{x-1}\rangle_N M[t_x\rangle M'$$

in which $M_0, \cdots, M_{x-1}, M \in \mathscr{L}_{(\xi,k)}$ and $M[t_x\rangle_N M' \notin \mathscr{L}_{(\xi,k)}$. By Proposition 2, there exists a unique firing sequence $\hat{t}_1 \cdots \hat{t}_{x-1} \in \hat{T}^*$, $\hat{t}_i \in \mathscr{T}(t_i)$ for $i = 1, \ldots, x-1$ such that $\hat{M}_0[\hat{t}_1\rangle_{\hat{N}} \hat{M}_1 \cdots \hat{M}_{x-1}[\hat{t}_y\rangle_{\hat{N}} \hat{M}$, $\hat{M}_{\uparrow P} = M$, and $\hat{M}(p_s) = k - \xi(M)$ holds.

By the proof of Proposition 2, at $\hat{M}$ there exists a unique transition of t_x, say $t_x^{r_1\cdots r_s}$, which is not disabled by the remainder places Q_i's. However, since $M' \notin \mathscr{L}_{(\xi,k)}$, we have $M(p_s) - \hat{C}(p_s, t_x^{r_1\cdots r_s}) = k - \xi(M) - (\xi(M') - \xi(M)) = k - \xi(M') < 0$. It indicates that $t_x^{r_1\cdots r_s}$ is disabled by p_s at $\hat{M}$ in $\hat{N}$. As a result, no transition $\hat{t} \in \mathscr{T}(t_x)$ is enabled at $\hat{M}$ to yield a marking $\hat{M}'$ such that $\hat{M}'_{\uparrow P} \notin \mathscr{L}_{(\xi,k)}$. Hence $R(\hat{N}, \hat{M}_0)_{\uparrow P} \subseteq \mathscr{L}_{(\xi,k)}$ holds. □ ■

A desirable property of the closed-loop $\langle \hat{N}, \hat{M}_0\rangle$ is *maximal permissiveness*, that is, if a marking evolution trajectory in the plant net is legal, such evolution should not be disabled by the control structure. This property ensures that the evolution of the closed-loop net is minimally restricted.

The maximal permissiveness of a closed-loop net for language specifications is often characterized in terms of supremal controllable sublanguages of automata [29] and Petri nets [10] when the supervisor is described as an external control agent. In the approach of this work, the action of the supervisor is embedded in the closed-loop net by means of both control places and duplicate transitions. For this particular reason, we want to clearly state the definition of maximally permissiveness as follows.

Definition 7 *Given a Petri net $\langle N, M_0\rangle$ and a legal marking set $\mathscr{L}$, a firing sequence $\sigma = t_1 t_2 \cdots t_{x-1} t_x \in T^*$ is* legal *if $M_0[t_1\rangle M_1 \cdots M_{x-1}[t_x\rangle M_x$ and $M_i \in \mathscr{L}$ for $i \in \{0, 1, \ldots, x\}$.* △

Definition 8 *Given a Petri net $\langle N, M_0\rangle$ and a legal marking set $\mathscr{L}$, a closed-loop net $\langle \hat{N}, \hat{M}_0\rangle$ with $\hat{N} = (P \cup P_S, \hat{T}, \hat{Pre}, \hat{Post})$ and $\hat{T} = \bigcup_{t \in T} \mathscr{T}(t)$ (i.e., it consists of only duplicate transitions) is* maximally permissive *if $\forall \sigma = t_1 t_2 \cdots t_{x-1} t_x \in T^*$ that is legal, $\exists \hat{t}_1, \ldots, \hat{t}_x$ such that $\hat{M}_0[\hat{t}_1 \hat{t}_2 \cdots \hat{t}_x\rangle$, where $\hat{t}_i \in \mathscr{T}(t_i)$.* △

The maximal permissiveness requires that if a firing sequence in the plant net is legal, i.e., its firing does not yield any intermediate illegal markings, then in the closed-loop net a firing sequence composed by corresponding duplicate transitions should be firable. Now we prove that the closed-loop net $\langle \hat{N}, \hat{M}_0\rangle$ obtained by the proposed procedure is maximally permissive, i.e., it prevents only those illegal trajectories of the plant.

Theorem 2 *Given a Petri net $\langle N, M_0\rangle$ and a stair-GMEC (ξ, k) with $\xi(M) = \sum_{i=1}^{s} f_i(M)$ and $f_i(M) = b_i \cdot \lfloor \mathbf{a}_i^T \cdot M / c_i \rfloor$, the closed-loop net $\langle \hat{N}, \hat{M}_0\rangle$ obtained by Algorithm 1 is maximally permissive by Definition 8.*

Proof: This theorem follows directly from Proposition 2. For a transition t satisfying $M_0[t\rangle_N M_1 \in \mathscr{L}$, there exists a unique $\hat{t} \in \mathscr{T}(t)$ such that $\hat{M}_0[\hat{t}\rangle_{\hat{N}} \hat{M}_1$ where $\hat{M}_1$ satisfies the three conditions in Proposition 2. This reasoning can be repeatedly applied by letting M_1 and $\hat{M}_1$ be the new initial marking of N and $\hat{N}$, respectively. Hence for any legal firing trajectory $M_0[t_1\rangle_N M_1 \cdots M_{x-1}[t_x\rangle_N M_x$, there exists a unique firing sequence $\hat{t}_1 \cdots \hat{t}_x \in \hat{T}^*$ such that $M_0[\hat{t}_1 \cdots \hat{t}_x\rangle_{\hat{N}}$. This indicates the maximal permissiveness of $\langle \hat{N}, \hat{M}_0\rangle$. □ ■

5 Complexity Analysis

In this section we discuss the complexity of the proposed control structure and compare it with the approach in literature.

5.1 Complexity of the Closed-loop Net

The proposed control structure to enforce a stair-GMEC (ξ, k) requires to add in the closed-loop net a unique residue place p_s and c_i remainder places for each f_i. Hence there will be $1 + \sum_{i=1}^{s} c_i$ places in the control structure of the closed-loop net. For a transition t and an f_i, if $\eta(f_i, t) \mod c_i \neq 0$ then c_i duplicate transitions are added in the closed-loop net. Therefore given a stair-GMEC (ξ, k) with $\xi(M) = f_1(M) + \cdots + f_n(M)$, a transition t will be duplicated to $\prod_{i=1}^{s} c_i$ transitions in the worst case in which the firing of t affects the value of all f_i's and $\eta(f_i, t) \mod c_i \neq 0$ holds for all f_i's. Hence the number of transitions in the closed-loop net will be $|\hat{T}| = |T| \cdot \prod_{i=1}^{s} c_i$. In conclusion, the total numbers of places and transitions in the closed-loop net are $|P| + \sum_{i=1}^{s} c_i + 1$ and $|T| \cdot \prod_{i=1}^{s} c_i$, respectively. Although the number of transitions grows up exponentially with the increase of the number of floor functions f_i in the worst case, in practice the number of duplications of a transition t would be less than $\prod_{i=1}^{s} c_i$ since not all transitions are duplicated at each iteration but only those whose firing affects the remainder of $\mathbf{a}_i^T \cdot M/c_i$. i.e., t is duplicated only if $\eta(f_i,t) \mod c_i \neq 0$.

5.2 Comparison with [7]

Now let us briefly compare in terms of structural complexity the controllers obtained by the proposed approach and by [7]. In both approaches duplicate transitions are introduced to realize the nonlinearity of the constraint. However, for nonlinear stair-GMECs the advantages of the proposed approach are twofold.

The work of [7] focuses on general types of nonlinear constraints in bounded Petri nets. Their approach seeks for a solution based on P-invariants and complementary places. Hence a full enumeration of all possible value changes from x to y of a nonlinear function f_i by firing a transition t is needed. This mechanism requires that each nonlinear function f_i appears in ξ is a function of only one bounded place p_i. On the contrary, since in this approach for stair-GMECs we do not need to track all possible value changes of ξ: only the remainder of $\mathbf{a}_i^T \cdot M/c_i$ for each f_i is recorded. Since the complementary place is not needed, this approach can be applied to unbounded nets.

Moreover, although in both approaches the number of duplicate transitions is exponential in the worst case, we note that the number of duplicate transitions in this approach (which is $\prod_{i=1}^{s} c_i$) is usually much less than that of [7] (which is $\prod_{i=1}^{s} |\hat{T}^{z_i}|$). Since in [7] $|\hat{T}^{z_i}|$ duplicate transitions are added for a plant transition t, each of which represents a possible value change of f_i from x to y by firing t. Hence $|\hat{T}^{z_i}| = K_{max} - K_{min} - |\eta(f_i,t)| + 1$ where K_{max} and K_{min} denote the upper and the lower bounds of $f_i(M)$, $M \in R(N,M_0)$, respectively. In general $|\hat{T}^{z_i}|$ is much larger than c_i and hence $\prod_{i=1}^{s} c_i \ll \prod_{i=1}^{s} |\hat{T}^{z_i}|$.

For example, if we apply the method in [7] to the net in Figure 3(a) and the stair-GMEC $\xi(M) = f_1(M) + f_2(M) = 2\lfloor M(p_1)/3 \rfloor - \lfloor M(p_2)/1 \rfloor \leq 14$, for f_1 there are 13 and 12 duplicate transitions of t_1 and t_2, respectively. However, as already shown in Example 3, for function f_1 transition t_1 is only duplicated to $c_1 = 3$ transitions while transition t_2 does not need to be duplicated. Moreover, since $|\hat{T}^{z_i}|$ grows with the increase of the bound of place p_i, for unbounded nets the method in [7] cannot be applied since $\hat{T}^{z_i}$ is not finite. A comprehensive example is also given in the next section.

6 Example

The Petri net in Figure 5 models the assembly system illustrated in Figure 4. Two types of parts A and B arrive (t_1 and t_2) at Workplace 1 and Workplaces 2 (p_1 and p_2), respectively, and then assembled by a robot (t_7). Once the robot t_7 is shut down, the remaining parts in workplaces p_1 and p_2 are automatically transported to

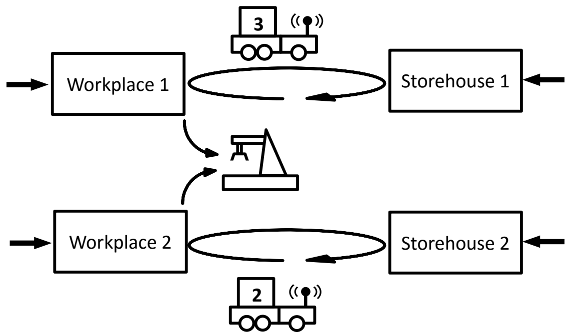

Figure 4: An assembly system for the example in Section 6.

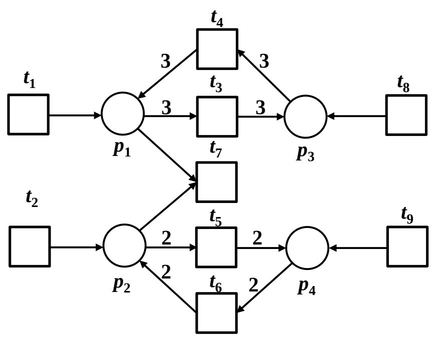

Figure 5: The Petri net model of Figure 4.

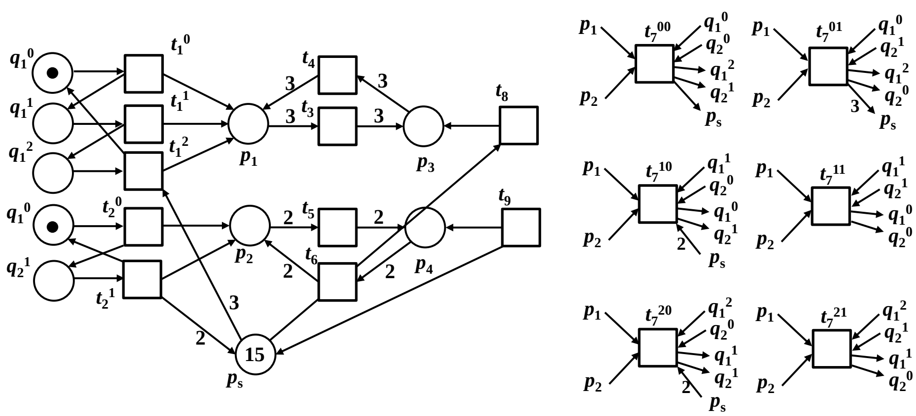

Figure 6: The closed-loop net of the net in Figure 5.

Storehouse 1 and Storehouse 2 (p_3 and p_4) by two AGVs (t_3 and t_5) that have different capacities, respectively. When the robot (t_7) recovers to function, the two AGVs automatically transport parts from storehouses back to the corresponding workplaces (t_4 and t_6). The parts can also arrive (t_8 and t_9) in the storehouses p_3 and p_4 from elsewhere.

Now we want to enforce a control policy that, once the robot (t_7) is shut down, after the transportation from the workplaces to the storehouses, the quantity of type A parts in Storehouse 1 (p_3) should not exceed the quantity of type B parts in Storehouse 2 (p_4) by more than 15 units. Hence the legal marking set can be defined by the following stair-GMEC (ξ, k):

$$3\left\lfloor \frac{M(p_1)}{3} \right\rfloor - 2\left\lfloor \frac{M(p_2)}{2} \right\rfloor + M(p_3) - M(p_4) \leq 15 \tag{10}$$

The Unbounded Case: By applying Algorithm 1 in this paper we can obtain the closed-loop net shown in Figure 6 which is maximally permissive. On the other hand, since the net is unbounded and $\mathscr{L}_{(\xi,k)}$ cannot be defined by an OR-AND GMEC, the methods in [7] and [22] cannot be applied.

The Bounded Case: Let us assume that the net has a bound $K = 15$ for all places, i.e., $R(N, M_0) = \{M \in \mathbb{N}^4 \mid M(p_i) \leq 15, i \in \{1,2,3,4\}\}$. In such a case $R(N, M_0)$ consists of 65,536 markings among which 62,758 markings are legal. Now let us compare the controller based on OR-AND GMECs [22], the nonlinear controller synthesis approach in [7], and Algorithm 1 proposed in this paper.

1. *[OR-AND GMEC Controller by [22]]* First, since the legal marking set is finite, there exists an OR-AND GMEC that defines it. Hence the method in [22] can apply. However, one can readily verify that such an OR-AND GMEC is too complex to be obtained: there is no efficient algorithm to recognize it from a linear-list of 62,758 legal markings. As a result, it is not possible to design a controller based on OR-AND GMECs. Moreover, once the value of K and/or k changes, the corresponding OR-AND GMEC has to be recalculated and the controller has to be completely recomputed.

2. *[Nonlinear Controller by [7]]* Second, since the net is bounded, the method in [7] can be applied, and the number of duplicate transitions in the resulting closed-loop net are listed in Table 1. The closed-loop net contains more than 300 transitions and hence is not presented graphically. In fact, the number of duplicate transitions grows with the increase of the value of K.

3. *[Controller by Algorithm 1]* Finally, by Algorithm 1 the closed-loop net is exactly the one in Figure 6. It contains only 17 transitions (details are also listed in Table 1) which is much less than that of [7]. In addition, this closed-loop net is maximally permissive: all 62,758 legal plant markings are reachable. Furthermore, this control structure is always the same regardless the value of the bound K.

Table 1: The number of duplicate transitions for the plant in Figure 5 by different methods.

Transitions	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9
Algorithm 1	3	2	1	1	1	1	6	1	1
[7]	15	15	13	13	14	14	225	1	1

7 Conclusion

In this paper a type of nonlinear marking specifications in Petri nets, called stair-GMECs, is proposed, which have a higher modeling power than classical GMECs. Two algorithms are developed to enforce a stair-GMEC as a closed-loop net, in which the control structure is composed by a residue counter, remainder counters, and duplicate transitions. The proposed control structure is maximally permissive. This approach is applicable to both bounded and unbounded nets. Our future topic is to extend this work to Petri nets with uncontrollable and unobservable transitions, and to explore properties of stair-GMECs with OR and AND relations.

References

[1] F. Basile, C. Carbone, and P. Chiacchio. Feedback control logic for backward conflict free choice nets. *IEEE Transactions on Automatic Control*, 52(3):387–400, 2007.

[2] F. Basile, R. Cordone, and L. Piroddi. Integrated design of optimal supervisors for the enforcement of static and behavioral specifications in Petri net models. *Automatica*, 49(11):3432–3439, 2013.

[3] F. Basile, R. Cordone, and L. Piroddi. A branch and bound approach for the design of decentralized supervisors in Petri net models. *Automatica*, 52:322–333, 2015.

[4] M. P. Cabasino, A. Giua, C. N. Hadjicostis, and C. Seatzu. Fault model identification and synthesis in Petri nets. *Discrete Event Dynamic Systems*, 25(13):419–440, 2015.

[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*, 47(5):1028–1034, 2011.

[6] Y. F. Chen, Z. W. Li, K. Barkaoui, and A. Giua. Optimal supervisory control of flexible manufacturing systems by Petri nets: A set classification approach. *IEEE Transactions on Automation Science and Engineering*, 11(2):549–563, 2014.

[7] Y. F. Chen, Z. W. Li, K. Barkaoui, and A. Giua. On the enforcement of a class of nonlinear constraints on Petri nets. *Automatica*, 55:116–124, 2015.

[8] A. Giua. Supervisory control of Petri nets with language specifications. In C. Seatzu, M. Silva, and J.H. van Schuppen, editors, *Control of discrete-event systems*, volume 433, pages 235–255. Springer, London, 2013.

[9] A. Giua, F. DiCesare, and M. Silva. Generalized mutual exclusion constraints for Petri nets with uncontrollable transitions. In *Proceedings of the IEEE Int. Conf. on Systems, Man, and Cybernetics*, pages 947–949, Chicago, USA, 1992.

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

[11] M. V. Iordache and P. J. Antsaklis. Supervision based on place invariants: A survey. *Discrete Event Dynamic Systems*, 16(4):4451–4492, 2006.

[12] M. V. Iordache and P. J. Antsaklis. Petri net supervisors for disjunctive constraints. In *Proceedings of the 26th American Control Conference*, pages 4951–4956, New York, USA, 2007.

[13] M. V. Iordache, P. Wu, F. Zhu, and P. J. Antsaklis. Efficient design of Petri-net supervisors with disjunctive specifications. In *Proceedings of the IEEE Int. Conf. on Automation Science and Engineering*, pages 936–941, Madison, USA, 2013.

[14] 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*, 42(4):437–462, 2012.

[15] 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*, 34(1):38–51, 2004.

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

[17] J. L. Luo, H. J. Ni, W. M. Wu, S. G. Wang, and M. C. Zhou. Simultaneous reduction of Petri nets and linear constraints for efficient supervisor synthesis. *IEEE Transactions on Automatic Control*, 60(1):88–103, 2015.

[18] J. L. Luo and K. Nonami. Approach for transforming linear constraints on Petri nets. *IEEE Transactions on Automatic Control*, 56(12):2751–2765, 2011.

[19] J. L. Luo, H. Shao, K. Nonami, and F. J. Jin. Maximally permissive supervisor synthesis based on a new constraint transformation method. *Automatica*, 48(6):1097–1101, 2012.

[20] J. L. Luo, W. M. Wu, H. Y. Su, and J. Chu. Supervisor synthesis for enforcing a class of generalized mutual exclusion constraints on Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 39(6):1237–1246, 2009.

[21] Z. Y. Ma, Z. W. Li, and A. Giua. Comments on "Maximally permissive supervisor synthesis based on a new constraint transformation method" [Automatica 48 (2012), 1097-1101]. *Automatica*, 51:131–134, 2015.

[22] 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*, 60:1774–1785, 2015.

[23] Z. Y. Ma, Z. W. Li, and A. Giua. Characterization of admissible marking sets in Petri nets with conflicts and synchronizations. *IEEE Transactions on Automatic Control*, pages DOI:10.1109/TAC.2016.2585647, to appear, 2017.

[24] Z. Y. Ma, Y. Tong, Z. W. Li, and A. Giua. Basis marking representation of Petri net reachability spaces and its application to the reachability problem. *IEEE Transactions on Automatic Control*, page to appear, 2017.

[25] J. Moody and P. Antsaklis. Petri net supervisors for DES with uncontrollable and unobservable transitions. *IEEE Transactions on Automatic Control*, 45(3):462–476, 2000.

[26] A. Nazeem and S. Reveliotis. Maximally permissive deadlock avoidance for resource allocation systems with r/w-locks. *Discrete Event Dynamic Systems*, 25(1):31–63, 2015.

[27] M. Pocci, I. Demongodin, N. Giambiasi, and A. Giua. Synchronizing sequences on a class of unbounded systems using synchronized Petri nets. *Discrete Event Dynamic Systems*, 26(1):85–108, 2016.

[28] M. Qin, Z. W. Li, M. C. Zhou, M. Khalgui, and O. Mosbahi. Deadlock prevention for a class of Petri nets with uncontrollable and unobservable transitions. *IEEE Transactions on Systems, Man and Cybernetics Part A: Systems and Humans*, 42(3):727–728, 2012.

[29] P. J. Ramadge and W. M. Wonham. The control of discrete event systems. *Proceedings of IEEE*, 77(1):81–98, 1989.

[30] M. Uzam. On suboptimal supervisory control of Petri nets in the presence of uncontrollable transitions via monitor places. *International Journal of Advanced Manufacturing Technology*, 47(5):567–579, 2010.

[31] 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*, 37(3):362–371, 2007.

[32] S. G. Wang, C. Y. Wang, and M. C. Zhou. Design of optimal monitor-based supervisors for a class of Petri nets with uncontrollable transitions. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(5):1248–1255, 2013.