ISA Transactions 61 (2016) 129–140

Contents lists available at ScienceDirect

ISA Transactions

journal homepage: www.elsevier.com/locate/isatrans

Monitor design with multiple self-loops for maximally permissive supervisors

YuFeng Chen [a,c], ZhiWu Li [b,a,*], Kamel Barkaoui [c], Murat Uzam [d]

[a] School of Electro-Mechanical Engineering Xidian University No. 2 South Taibai Road Xi'an 710071, China
[b] Institute of Systems Engineering Macau University of Science and Technology Taipa, Macau 999078, China
[c] Cedric Lab and Computer Science Department Conservatoire National des Arts et Métiers, Paris 75141, France
[d] Elektrik-Elektronik Muhendisliği Bölümü, Muhendislik-Mimarlik Fakültesi Meliksah Universitesi, 38280 Talas/Kayseri, Turkey

ARTICLE INFO

Article history:
Received 11 October 2014
Received in revised form
22 September 2015
Accepted 26 December 2015
Available online 5 February 2016
This paper was recommended for publication by Dr. Q.-G. Wang

Keywords:
Petri net
Flexible manufacturing system (FMS)
Deadlock prevention
Supervisory control
Self-loop

ABSTRACT

In this paper, we improve the previous work by considering that a control place can have multiple self-loops. Then, two integer linear programming problems (ILPPs) are formulated. Based on the first ILPP, an iterative deadlock control policy is developed, where a control place is computed at each iteration to implement as many marking/transition separation instances (MTSIs) as possible. The second ILPP can find a set of control places to implement all MTSIs and the objective function is used to minimize the number of control places. It is a non-iterative deadlock control strategy since we need to solve the ILPP only once. Both ILPPs can make all legal markings reachable in the controlled system, i.e., the obtained supervisor is behaviorally optimal. Finally, we provide examples to illustrate the proposed approaches.

1. Introduction

Petri nets [42] are an effective tool to model and control flexible manufacturing systems (FMSs) [56]. Petri nets have compact structures and can be represented in the form of matrixes. Thus, they can be simply analyzed by linear algebras. Deadlocks [16] are a constant issue in FMSs [9,14,17,31,33,34,36–38,40] due to the competition for limited shared resources. The last two decades have shown that Petri nets are widely used in the deadlock control of FMSs, which leads to abundant results [3–5,12,13,15,19,20,23,24,27–30,32,39,58,59,61,63–67,69].

In the framework of Petri nets, a deadlock prevention approach usually finds a supervisor for a system to be controlled, which consists of control places and the arcs connecting them to transitions that belong to the system. The supervisor is computed off-line and once a deadlock control policy is established and enforced, no deadlocks can occur anymore. Based on Petri nets, the performance of a deadlock control policy is always evaluated by three criteria: behavioral permissiveness, structural complexity, and computational complexity. Behavioral permissiveness is measured by the number of legal states kept in the controlled system. In the case that all legal states of a system to be controlled are reachable, the corresponding supervisor is said to be maximally permissive (or optimal). The structural complexity is defined in terms of the number of control places and arcs in a Petri net supervisor. A simplification of the supervisory structure can reduce the costs of hardware and software at implementation stage. An efficient computation indicates that the supervisor can be obtained in a reasonable time. In this work, we mainly focus on the optimization of the first two criteria.

Behavioral optimality plays an important role in the development of deadlock control of Petri nets. A representative work is the theory of regions proposed by Ghaffari et al. [25] and Uzam [52], which can definitely find a maximally permissive Petri net supervisor if it exists. The approach first generates the reachability graph of a net model. Then, the set of marking/transition separation instances (MTSIs) is derived. An MTSI is a pair of a marking M and a transition t, denoted as (M,t), where M is a legal marking and once t fires at M, it yields an illegal marking. For deadlock control purposes, Ghaffari et al. [25] develop an iterative approach where at each iteration, an MTSI (M,t) is singled out and a linear programming model is designed to find a control place to implement (M,t) by preventing t from firing at M. Meanwhile, all legal

* Corresponding author.
E-mail addresses: yfchen@mail.xidian.edu.cn (Y. Chen), zhwli@xidian.edu.cn (Z. Li), kamel.barkaoui@cnam.fr (K. Barkaoui), murat_uzam@meliksah.edu.tr (M. Uzam).

http://dx.doi.org/10.1016/j.isatra.2015.12.016
0019-0578/

markings are ensured to be reachable. The process carries out until all MTSIs are implemented. Then, a set of control places is obtained, which can make the controlled system live with all legal markings. That is to say, the obtained supervisor is behaviorally optimal. However, the approach does not consider the structural complexity since it always leads to a supervisor with too many control places.

Another representative study based on the reachability graph analysis is presented by Uzam and Zhou [53,54]. They classify a reachability graph into two parts: a live zone (LZ) and a deadlock zone (DZ), where the LZ contains all legal markings and the DZ contains all illegal markings. Then, the set of first-met bad markings (FBMs) is computed, where an FBM is an illegal marking that can be directly reached by firing a transition at a legal marking. That is to say, an FBM is within the DZ and represents the very first entry from the LZ to the DZ. Uzam and Zhou also develop an iterative approach to prevent deadlocks in a net model. An FBM is forbidden by using a place invariant (PI) based control place synthesis method [62]. Once all FBMs are forbidden, the controlled system is live. This approach is easy to use since it does not require to compute control places by solving linear programming problems. However, it cannot guarantee the behavioral optimality of the obtained supervisor. Motivated by Uzam and Zhou's work [54], the work in [6] develops a maximally permissive deadlock control method such that a control place is computed by solving an integer linear programming problem (ILPP) while an FBM is prohibited but no legal markings are forbidden. The process cannot terminate until all FBMs are forbidden. Then, we can obtain a supervisor with all legal markings reachable in the resulting controlled net model. Meanwhile, a marking reduction approach is proposed to reduce the number of markings that need to be considered. Thus, the number of constraints in the ILPP can be reduced. In [7,8], ILPPs are proposed to design optimal supervisors with the minimal number of control places. The work [7,8] can successfully handle the behavioral permissiveness and structural complexity problems but suffers from the computational complexity due to the existence of too many constraints in the ILPPs.

All the above studies based on Petri nets are in the framework of pure net models. However, there exist some Petri net models that cannot be optimally controlled by pure net supervisors [68]. Self-loops are a classical non-pure Petri net structure. A self-loop contains two arcs (p, t) and (t, p) connecting a place p and a transition t. Self-loops are used in several papers [18,42,48,54,55,57] for structural reductions, systems synthesis, system modeling, etc. In the previous work [11], self-loops are used to design maximally permissive supervisors to handle deadlock problems in FMSs. Similar to the theory of regions [25,52], we prevent deadlocks by implementing all MTSIs of a net to be controlled. For an MTSI (M, t), we assume that there is a self-loop associated with t in a control place. Then, an ILPP is formulated to design the control place. The constraints in the ILPP can ensure that t is disabled at M and no legal marking is forbidden. Meanwhile, no arcs in the LZ of the net model are removed by the self-loop in the control place. The objective function is used to maximize the number of t-critical MTSIs implemented by the control place, where a t-critical MTSI is the one with t as its paired transition. Experimental results in [11] show that Petri net supervisors with self-loops can optimally control the net models that cannot be optimally controlled by a pure net supervisor. Thus, Petri nets with self-loops are more powerful than pure nets in modeling and controlling FMSs.

In this paper, we aim to reduce the structural complexity of the previous work [11]. Instead of one self-loop in one control place, we allow multiple self-loops in a control place. We assume that a control place has a self-loop for each critical transition. A transition t is said to be critical if there exist MTSIs such that t is the paired transition. Then, we propose an ILPP to compute the control place that can implement as many MTSIs as possible. Thus, the obtained supervisor is structurally simple. We also proposed another ILPP to obtain all control places at a time. The constraints are designed to ensure that each MTSI is implemented by at least one control place and the objective function can minimize the number of control places. As a result, all control places can be obtained by solving only one ILPP and the obtained supervisor is structurally minimal in the sense of the number of control places. A drawback of the proposed methods is that they may fail to find a solution if the proposed ILPP for a given net model has no solution. In summary, we reach the following contributions in this work:

(1) In the case that a control place can have multiple self-loops, an ILPP is developed to design an optimal control place with a self-loop for each critical transition. The constraints in the ILPP are used to make all legal markings reachable in the controlled system and the objective function can ensure that the computed control place implements as many MTSIs as possible. Based on the proposed ILPP, an iterative deadlock prevention policy is developed, where a control place is designed at each iteration and the process carries out until all MTSIs are implemented. Hence, we can obtain an optimal supervisor with a small number of control places.
(2) In order to minimize the number of control places in the obtained supervisor, we formulate an ILPP to design a set of control places with self-loops and its objective function is used to select the minimal number of control places. As a result, we can obtain an optimal supervisor with the minimal number of control places. The proposed ILPP can lead to a non-iterative deadlock control policy since it can find all control places to implement all MTSIs by solving only one ILPP.
(3) The proposed approaches are applicable to all FMS-oriented classes of Petri net models in the literature, including PPN [1,26,60], S^3PR [2], ES^3PR [49], S^4PR [50], S^*PR [22], S^2LSPR [44], S^3PGR^2 [45], and S^3PMR [28].

The rest of this paper is organized as follows. Section 2 briefly recalls some basics of Petri nets. In Section 3, we propose two ILPPs to design optimal supervisors with compact structures and an illustrative example is presented to show the applications of the proposed methods in detail. Some examples from the literature are provided in Section 4 to show the control performance of the proposed methods. Finally, we conclude the paper in Section 5.

2. Preliminaries

This section only recalls the basics of Petri nets. More details can be found in [6,7,11,42].

A Petri net [42] is a four-tuple $N=(P, T, F, W)$ where P is a set of places and T is a set of transitions with $P \cap T = \varnothing$, and $F \subseteq (P \times T) \cup (T \times P)$ is a flow relation of the net. $W: (P \times T) \cup (T \times P) \rightarrow \mathbb{N}$ assigns a weight to an arc: $W(x, y) > 0$ if $(x, y) \in F$, and $W(x, y) = 0$, otherwise, where $x, y \in P \cup T$ and $\mathbb{N}$ is the set of non-negative integers. ${}^\bullet x = \{y \in P \cup T \mid (y, x) \in F\}$ is the preset of x and $x^\bullet = \{y \in P \cup T \mid (x, y) \in F\}$ the postset of x. A marking is a mapping $M: P \rightarrow \mathbb{N}$ where $M(p)$ denotes the number of tokens in place p. 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, denoted as $M[t\rangle$, if $\forall p \in {}^\bullet t$, $M(p) \geq W(p, t)$. Once a transition t is enabled at M and 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, M_0) is denoted by $R(N, M_0)$ and its reachability graph is denoted as $G(N, M_0)$. A transition $t \in T$ is live at M_0 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 at M_0. It is dead at M_0 if $\nexists t \in T$, $M_0[t\rangle$.

In this paper, some notations and definitions on the reachability analysis of Petri nets and the marking reduction technique are originally from [6,7,11,53]. Specifically, they can be found in [11]. In order to focus on the main contributions of this work and also thanks to the limited space, we do not provide them repeatedly and encourage the readers to find more details in [11].

3. Simplification of supervisory structures

The work in [11] presents an approach to design an optimal supervisor where a control place has a self-loop. In this section, we develop two ILPPs to improve the previous work by simplifying the supervisory structure where a control place can have multiple self-loops.

3.1. Design of a supervisor with a small number of control places

The constraint that includes enabled conditions can be found in [41,62]. In this paper, we consider the following constraint that includes enabled conditions for multiple transitions:

$$\sum_{i=1}^{n} l_i \cdot \mu_i + \sum_{j=1}^{m} w_j \cdot q_{t_j} \le \beta \tag{1}$$

where w_j is a nonnegative integer and $q_{t_j} \in \{0,1\}$ represents whether transition $t_j \in T$ is enabled. Specially, $q_{t_j} = 1$ indicates that t_j is enabled and $q_{t_j} = 0$ means that t_j is disabled.

Eq. (1) can be considered as two parts: (1) $\sum_{i=1}^{n} l_i \cdot \mu_i$ cannot be greater than β and (2) once $\sum_{i=1}^{n} l_i \cdot \mu_i$ is greater than $\beta - w_j$, t_j is disabled. For (1), it can be implemented by introducing a slack variable μ_s to form a PI, as shown in Section 3 of [11]. For (2), $\sum_{i=1}^{n} l_i \cdot \mu_i > \beta - w_j$ implies that $\mu_s < w_j$, i.e., t_j is disabled if $\mu_s < w_j$. In this case, for each transition t_j, we introduce a self-loop with a weight w_j between p_s and t_j, i.e., $W(p_s, t_j) = W(t_j, p_s) = w_j$. Then, t_j is disabled at a marking M if $M(p_s) < w$ and enabled if $M(p_s) \ge w$. Once t_j is enabled and fires, it does not change the tokens in p_s.

By Corollary 1 in [11], we only consider markings in the minimal covering set $\mathcal{M}_L^\star$ of legal markings (Definition 5 in [11]) to guarantee the reachability of all legal markings, i.e.,

$$\sum_{i \in \mathbb{N}_A} l_i \cdot M_l(p_i) \le \beta, \quad \forall M_l \in \mathcal{M}_L^\star \tag{2}$$

Now we recall some necessary results from [11] as follows.

Definition 1 (*Chen et al. [11]*). Let t be a critical transition (the set of critical transitions, see Eq. (16) in [11]) and (M, t) an MTSI. (M, t) is called a t-critical MTSI. The set of t-critical MTSIs is denoted as Ω_t.

Theorem 1 (*Chen et al. [11]*). *Let (M_1, t) and (M_2, t) be two MTSIs in Ω_t with $M_1 \le_A M_2$. If (M_1, t) is implemented by a control place with a self-loop associated with t, then (M_2, t) is implemented by the control place.*

Definition 2 (*Chen et al. [11]*). Let $\Omega_t^\star$ be a subset of t-critical MTSIs. $\Omega_t^\star$ is called a minimal covered set of t-critical MTSIs if the following two conditions are satisfied:

(1) $\forall (M, t) \in \Omega_t$, $\exists (M', t) \in \Omega_t^\star$ s.t. $M' \le_A M$;
(2) $\forall (M, t) \in \Omega_t^\star, \nexists (M'', t) \in \Omega_t^\star$ s.t. $M'' \le_A M$ and $M \ne M''$.

Corollary 1 (*Chen et al. [11]*). *If all MTSIs in $\Omega_t^\star$ are implemented by control places with self-loops associated with t, then all MTSIs in Ω_t are implemented by the control places.*

For optimal deadlock control purposes, we only consider to disable the critical transitions. Thus, we suppose that a control place p_s has a self-loop associated with each critical transition, i.e., $W(p_s, t_q) = W(t_q, p_s) = w_q$, $\forall t_q \in T_c$ (the set of critical transitions, see Eq. (16) in [11]). We define the minimal covered set of MTSIs as follows:

Definition 3. Let $\Omega^\star$ be a subset of MTSIs. $\Omega^\star$ is called a minimal covered set of MTSIs if the following two conditions are satisfied:

(1) $\forall (M, t) \in \Omega, \exists (M', t) \in \Omega^\star$ s.t. $M' \le_A M$;
(2) $\forall (M, t) \in \Omega^\star, \nexists (M'', t) \in \Omega^\star$ s.t. $M'' \le_A M$ and $M \ne M''$.

It is obvious that $\Omega^\star$ is the union of $\Omega_t^\star$ (the minimal covered set of t-critical MTSIs, Definition 8 in [11]) with $t \in T_c$, i.e.,

$$\Omega^\star = \bigcup_{t \in T_c} \Omega_t^\star \tag{3}$$

Corollary 2. *If all MTSIs in $\Omega^\star$ are implemented by a set of control places where each control place has a self-loop associated with each critical transition, then all MTSIs are implemented by the control places.*

Proof. It follows immediately from Corollary 1. □

For each critical transition $t_q \in T_c$, it should be enabled at each of its enabled good markings. Then, the enabled condition is

$$\sum_{i \in \mathbb{N}_A} l_i \cdot (M_k(p_i) + [N](p_i, t_q)) \le \beta - w_q, \quad \forall M_k \in \mathcal{E}_{t_q}^\star, \quad \forall t_q \in T_c \tag{4}$$

where $[N]$ is the incidence matrix of the net to be controlled and $\mathcal{E}_{t_q}^\star$ is the minimal covering set of t_q-enabled good markings (see Definition 6 in [11]).

We introduce a set of variables $f_{j,q}$'s ($\forall (M_j, t_q) \in \Omega^\star$) to represent whether an MTSI $(M_j, t_q) \in \Omega^\star$ is implemented by p_s. The disabled condition for (M_j, t_q) is

$$\sum_{i \in \mathbb{N}_A} l_i \cdot (M_j(p_i) + [N](p_i, t_q)) \ge -Q \cdot (1 - f_{j,q}) + \beta - w_q + 1, \quad \forall (M_j, t_q) \in \Omega^\star \tag{5}$$

where $f_{j,q} \in \{0,1\}$ ($\forall (M_j, t_q) \in \Omega^\star$). In Eq. (5), $f_{j,q} = 1$ indicates that (M_j, t_q) is implemented by p_s and $f_{j,q} = 0$ indicates that (M_j, t_q) is not implemented.

By combining Eqs. (2), (4) and (5), the following ILPP is developed to design an optimal control place p_s with a self-loop associated with each critical transition $t_q \in T_c$, which is denoted as the Implementation of Maximal Number of MTSIs (IMNM).

IMNM:

$$\max f = \sum_{(M_j, t_q) \in \Omega^\star} f_{j,q}$$

subject to

$$\sum_{i \in \mathbb{N}_A} l_i \cdot M_l(p_i) \le \beta, \forall M_l \in \mathcal{M}_L^\star \tag{6}$$

$$\sum_{i \in \mathbb{N}_A} l_i \cdot (M_k(p_i) + [N](p_i, t_q)) \le \beta - w_q, \quad \forall M_k \in \mathcal{E}_t^\star, \quad \forall t_q \in T_c \tag{7}$$

$$\sum_{i \in \mathbb{N}_A} l_i \cdot (M_j(p_i) + [N](p_i, t_q)) \ge -Q \cdot (1 - f_{j,q}) + \beta - w_q + 1, \quad \forall (M_j, t_q) \in \Omega^\star \tag{8}$$

$l_i \in \{0, 1, 2, \ldots\}, \quad i \in \mathbb{N}_A$
$\beta \in \{0, 1, 2, 3, \ldots\}$

Table 1
The numbers of constraints and variables in IMNM.

Eq.	No. constraints	Variable	No. variables
(6)	$\lvert\mathcal{M}_L^\star\rvert$	l_i	$\lvert P_A\rvert$
(7)	$\sum_{t_q\in T_c}\lvert\mathcal{E}_{t_q}^\star\rvert$	$f_{j,q}$	$\lvert\Omega^\star\rvert$
(8)	$\lvert\Omega^\star\rvert$	w_q	$\lvert T_c\rvert$
		β	1
Total	$\lvert\mathcal{M}_L^\star\rvert+\sum_{t_q\in T_c}\lvert\mathcal{E}_{t_q}^\star\rvert+\lvert\Omega^\star\rvert$	Total	$\lvert P_A\rvert+\lvert\Omega^\star\rvert+\lvert T_c\rvert+1$

$$w_q\in\{0,1,2,3,\ldots\},\quad \forall t_q\in T_c$$
$$f_{j,q}\in\{0,1\},\quad \forall(M_j,t_q)\in\Omega^\star$$

The objective function f is used to maximize the number of MTSIs that are implemented by control place p_s. Denote its optimal value by f^*. If $f^*=0$, we have $\forall(M_j,t_q)\in\Omega^\star$, $f_{j,q}=0$, implying that no MTSI can be implemented by p_s. In this case, the proposed method cannot obtain an optimal control place with a self-loop associated with each critical transition. If $f^*=\lvert\Omega^\star\rvert$, we have $\forall(M_j,t_q)\in\Omega^\star$, $f_{j,q}=1$, implying that only one control place with a self-loop associated with t is required to implement all MTSIs. The number of constraints and variables in IMNM is summarized in Table 1.

Algorithm 1 first computes the reachability graph of a Petri net. Markings in $\mathcal{M}_L^\star$ are considered only to keep all legal markings reachable. For a transition t_q in T_c, only markings in $\mathcal{E}_{t_q}^\star$ are considered to satisfy the enabled conditions. Furthermore, we only consider the MTSIs in $\Omega^\star$. Thus, the reduction techniques can greatly reduce the number of constraints in IMNM to compute control places.

Algorithm 1. Computation of an optimal supervisor with self-loops.

Input: Petri net model (N,M_0) of an FMS with $N=(P^0\cup P_A\cup P_R,T,F,W)$.
Output: an optimally controlled Petri net system (N^α,M_0^α).
1. Compute the set $\mathcal{M}_L$ of legal markings and the set Ω of MTSIs for (N,M_0).
2. Compute the set T_c of critical transitions and the minimal covering set $\mathcal{M}_L^\star$ of legal markings for (N,M_0).
3. V_M:=∅./* V_M is used to denote the set of control places to be computed.*/
4. **foreach** $t\in T_c$ **do**
   Compute the minimal covering set $\mathcal{E}_t^\star$ of t-enabled good markings and the minimal covered set $\Omega^\star$ of MTSIs.
5. **while** $\{\Omega^\star\neq\varnothing\}$ **do**
   Solve IMNM. If $f^*=0$, exit, as no control place can be obtained.
   Let l_i's $(i\in\mathbb{N}_A)$, β, and w_q's $(t_q\in T_c)$ be the solution.
   Compute control place p_s with a self-loop associated with each critical transition.
   V_M:=$V_M\cup\{p_s\}$ and $\Omega^\star$:=$\Omega^\star-\Omega_{p_s}$ (the set Ω_{p_s} of MTSIs implemented by p_s, Definition 9 in [11]).
   endwhile
6. Add all control places in V_M to (N,M_0) and denote the resulting net system as (N^α,M_0^α).
7. Output (N^α,M_0^α).
8. End.

Now we discuss the computational complexity of Algorithm 1. First, it is based on reachability graph analysis of a Petri net model, which suffers from the state explosion problem since the number of reachable markings increases exponentially with respect to the size of a Petri net model. Second, we need to solve ILPPs, which are NP-hard in theory. However, the computational time to solve an ILPP greatly depends on the number of constraints and variables. Thus, we propose a marking reduction technique to reduce the number of constraints in the ILPPs, aiming to reduce the computational burden of the proposed method. In summary, the computational complexity of the proposed method is still NP-hard.

Theorem 2. *Algorithm 1 can obtain a maximally permissive net supervisor for a Petri net model of an FMS if there exists a solution that satisfies Eqs. (2), (4) and (5) for each MTSI in $\Omega^\star$.*

Proof. According to Corollary 1 in [11], Eq. (2) can ensure that each computed control place does not forbid any legal marking. By Corollary 2 in [11], if a transition $t_q\in T_c$ is enabled at a good marking, Eq. (4) is used to ensure that the additional self-loop does not disable its firing. By Corollary 2, if all MTSIs in $\Omega^\star$ are implemented, all MTSIs are implemented. That is to say, the final controlled system is live and all legal markings are kept, i.e., the conclusion holds. □

3.2. Design of a supervisor with the minimal number of control places

In this section, we aim to compute an optimal supervisor with the minimal number of control places. We suppose that there are $n_c(n_c\geq 2)$ control places to be computed, where each control place has a self-loop associated with each critical transition. Then, we propose an ILPP to select the minimal number of control places from the n_c control places to implement all MTSIs in $\Omega^\star$. Let p_{s_1}, p_{s_2}, …, and $p_{s_{n_c}}$ be the n_c control places where $\forall p_{s_z}$ $(z\in\{1,2,\ldots,n_c\})$ and $\forall t_q\in T_c$, the weight on the self-loop between p_{s_z} and t_q is $w_{z,q}$, i.e., $W(p_{s_z},t_q)=W(t_q,p_{s_z})=w_{z,q}$. Meanwhile, each control place p_{s_z} is used to implement the condition:

$$\sum_{i\in\mathbb{N}_A} l_{z,i}\cdot\mu_i\leq\beta_z \tag{9}$$

In this case, the reachability condition, Eq. (2), is modified as:

$$\sum_{i\in\mathbb{N}_A} l_{z,i}\cdot M_l(p_i)\leq\beta_z,\ \forall M_l\in\mathcal{M}_L^\star,\quad\forall z\in\{1,2,\ldots,n_c\} \tag{10}$$

The enabled conditions for the markings in $\mathcal{E}_{t_q}^\star(\forall t_q\in T_c)$ are presented as:

$$\sum_{i\in\mathbb{N}_A} l_{z,i}\cdot(M_k(p_i)+[N](p_i,t_q))\leq\beta_z-w_{z,q},\quad\forall M_k\in\mathcal{E}_{t_q}^\star,$$
$$\forall t_q\in T_c,\quad\forall z\in\{1,2,\ldots,n_c\} \tag{11}$$

We introduce a set of variables $f_{z,j,q}$'s $(\forall(M_j,t_q)\in\Omega^\star,\forall z\in\{1,2,\ldots,z\})$ to represent whether an MTSI $(M_j,t_q)\in\Omega^\star$ is implemented by p_{s_z}. The disabled condition for (M_j,t_q) is modified as:

$$\sum_{i\in\mathbb{N}_A} l_{z,i}\cdot(M_j(p_i)+[N](p_i,t_q))\geq -Q$$
$$\cdot(1-f_{z,j,q})+\beta_z-w_{z,q}+1,\quad\forall(M_j,t_q)\in\Omega^\star,\quad\forall z\in\{1,2,\ldots,n_c\} \tag{12}$$

where $f_{z,j,q}\in\{0,1\}(\forall(M_j,t_q)\in\Omega^\star,\forall z\in\{1,2,\ldots,z\})$. In Eq. (12), $f_{z,j,q}=1$ indicates that (M_j,t_q) is implemented by p_{s_z} and $f_{z,j,q}=0$ indicates that (M_j,t_q) is not implemented by p_{s_z}. For each MTSI $(M_j,t_q)\in\Omega^\star$, it should be implemented by at least one control place. Hence, we have

$$\sum_{z=1}^{n_c} f_{z,j,q}\geq 1,\quad\forall(M_j,t_q)\in\Omega^\star \tag{13}$$

We introduce a binary variable $h_z\in\{0,1\}$ for each control place p_{s_z} to represent whether p_{s_z} is selected or not. The fact h_z=1 indicates that p_{s_z} is selected and h_z=0 implies that p_{s_z} is not selected. If a control place p_{s_z} is not selected, it cannot implement

Table 2
The numbers of constraints and variables in MNCP.

Eq.	No. constraints	Variable	No. variables
(16)	$n_c \cdot \|\mathcal{M}_L^\star\|$	$l_{z,i}$	$n_c \cdot \|P_A\|$
(17)	$n_c \cdot \sum_{t_q \in T_c} \|\mathcal{E}_{t_q}^\star\|$	$f_{z,j,q}$	$n_c \cdot \|\Omega^\star\|$
(18)	$n_c \cdot \|\Omega^\star\|$	h_z	n_c
(19)	$\|\Omega^\star\|$	β_z	n_c
(20)	$n_c \cdot \|\Omega^\star\|$	$w_{z,q}$	$n_c \cdot T_c$
Total	$n_c \cdot (\|\mathcal{M}_L^\star\| + \sum_{t_q \in T_c} \|\mathcal{E}_{t_q}^\star\| + 2\|\Omega^\star\|) + \|\Omega^\star\|$	Total	$n_c \cdot (\|P_A\| + \|\Omega^\star\| + \|T_c\| + 2)$

any MTSI. Therefore, we have

$$f_{z,j,q} \le h_z, \quad \forall (M_j, t_q) \in \Omega^\star, \quad \forall z \in \{1, 2, \ldots, n_c\} \tag{14}$$

Finally, an objective function is used to select the minimal number of control places to implement all MTSIs, as follows:

$$\min h = \sum_{z=1}^{n_c} h_z \tag{15}$$

Combining Eqs. (10)–(15), we can obtain an ILPP, namely the Minimization of the Number of Control Places (MNCP), as presented below:

MNCP:

$$\min h = \sum_{z=1}^{n_c} h_z$$

subject to

$$\sum_{i \in \mathbb{N}_A} l_{z,i} \cdot M_l(p_i) \le \beta_z, \forall M_l \in \mathcal{M}_L^\star, \quad \forall z \in \{1, 2, \ldots, n_c\} \tag{16}$$

$$\sum_{i \in \mathbb{N}_A} l_{z,i} \cdot (M_k(p_i) + [N](p_i, t_q)) \le \beta_z - w_{z,q}, \quad \forall M_k \in \mathcal{E}_{t_q}^\star, \forall t_q \in T_c, \forall z \in \{1, 2, \ldots, n_c\} \tag{17}$$

$$\sum_{i \in \mathbb{N}_A} l_{z,i} \cdot (M_j(p_i) + [N](p_i, t_q)) \ge -Q \cdot (1 - f_{z,j,q}) + \beta_z - w_{z,q} + 1, \forall (M_j, t_q) \in \Omega^\star, \quad \forall z \in \{1, 2, \ldots, n_c\} \tag{18}$$

$$\sum_{z=1}^{n_c} f_{z,j,q} \ge 1, \forall (M_j, t_q) \in \Omega^\star \tag{19}$$

$$f_{z,j,q} \le h_z, \forall (M_j, t_q) \in \Omega^\star, \quad \forall z \in \{1, 2, \ldots, n_c\} \tag{20}$$

$$l_{z,i} \in \{0, 1, 2, \ldots\}, i \in \mathbb{N}_A, \quad \forall z \in \{1, 2, \ldots, n_c\}$$
$$\beta_z \in \{0, 1, 2, 3, \ldots\}, \quad \forall z \in \{1, 2, \ldots, n_c\}$$
$$h_z \in \{0, 1\}, \quad \forall z \in \{1, 2, \ldots, n_c\}$$
$$w_{z,q} \in \{0, 1, 2, 3, \ldots\}, \quad \forall t_q \in T_c, \quad \forall z \in \{1, 2, \ldots, n_c\}$$
$$f_{z,j,q} \in \{0, 1\}, \quad \forall (M_j, t_q) \in \Omega^\star, \quad \forall z \in \{1, 2, \ldots, n_c\}$$

The objective function h is used to minimize the number of control places to be computed. Denote its optimal solution as h^*. If MNCP has an optimal solution, then we can claim that the given Petri net model can be optimally controlled by the minimal number h^* of control places with self-loops, as stated below.

Theorem 3. *Given a Petri net model, if MNCP has an optimal solution h^*, then we can obtain an optimal supervisor that has the minimal number h^* of control places with self-loops.*

Proof. According to Eq. (16), every computed control place does not forbid any legal marking. For each critical transition, Eq. (17) ensures that it is enabled at all its enabled good markings. By Eqs. (18) and (19), every MTSI in $\Omega^\star$ is implemented by at least one selected control place. Thus, all MTSIs are accordingly implemented. Eq. (20) ensures that only the selected control places are used to implement MTSIs. The objective function ensures that the minimal number of control places is selected, where the number of control places in the obtained supervisor is equal to the optimal solution h^*. In summary, the conclusion holds. □

If the given n_c is too small, i.e., it is smaller than the minimal number of control places for the given net model, MNCP has no solution. In this case, we can rewrite MNCP by increasing n_c. In fact, the greatest number of control places is such that we design a control place for each MTSI in $\Omega^\star$. Thus, we have $n_c \le |\Omega^\star|$. If MNCP has no solution with $n_c = |\Omega^\star|$, it fails to design an optimal supervisor with self-loops for the given Petri net model.

Next, we discuss MNCP from the viewpoint of the numbers of constraints and variables, as shown in Table 2. It can be seen that both instances greatly depend on n_c. Suppose that there exists an optimal supervisor with the minimal number of control places for a given Petri net model and denote the number as n_c^*. If n_c is too small, i.e., $n_c < n_c^*$, MNCP has no solution. In this case, we have to increase n_c and solve MNCP again. Therefore, the proper range for n_c is from n_c^* to $|\Omega^\star|$. The best case is that n_c initially happens to be n_c^*. Then an optimal supervisor with a minimal structure can be obtained and MNCP has a small number of constraints and variables. However, given a net model, we do not know the exact value of n_c^* before solving MNCP. That is to say, it is hard to set n_c initially. In fact, the closer the given n_c is to n_c^*, the more efficient the proposed method is. In the following, we provide three methods to give an initial value for n_c.

First, we can initially assign n_c as 1. If $n_c < n_c^*$, MNCP has no solution. Then, we increase n_c and solve MNCP again. The process carries out until $n_c = n_c^*$. Then, MNCP has an optimal solution. By using this method, the number of iterations is n_c^*.

Second, we can propose an initial value by other existing effective methods and make it in the range from n_c^* to $|\Omega^\star|$. For instance, in [11], we propose an iterative approach to obtain an optimal supervisor with self-loops. The approach is rather efficient though it cannot guarantee the minimality of the supervisory structure. Suppose that the method proposed in [11] obtains a supervisor with n_c' control places. Then, we have $n_c^* \le n'$. Just as stated in [11], n_c' is very close to n_c^*. Thus, we can initially set $n_c = n_c'$.

Third, we can assign $n_c = |T_c|$. The work in [11] shows that the number of control places is equal to $|T_c|$ in the best case. Thus, without the solution of any other work, we can initially make $n_c = |T_c|$. If $n_c^* \le |T_c|$, MNCP has a solution with $n_c = |T_c|$. Otherwise, we can increase n_c and solve MNCP again. In this case, the number of iteration is greatly reduced and MNCP becomes effective and efficient.

3.3. An illustrative example

In this section, the Petri net model of an FMS is used to illustrate the proposed approaches. In the following, for the sake of simplicity, we introduce a multiset formalism to represent the markings in $\mathcal{M}_L^\star$ and $\mathcal{E}_t^\star$. Since only operation places are considered, we use $\sum_{i \in \mathbb{N}_A} M(p_i) p_i$ to denote marking M in $\mathcal{M}_L^\star$ or $\mathcal{E}_t^\star$.

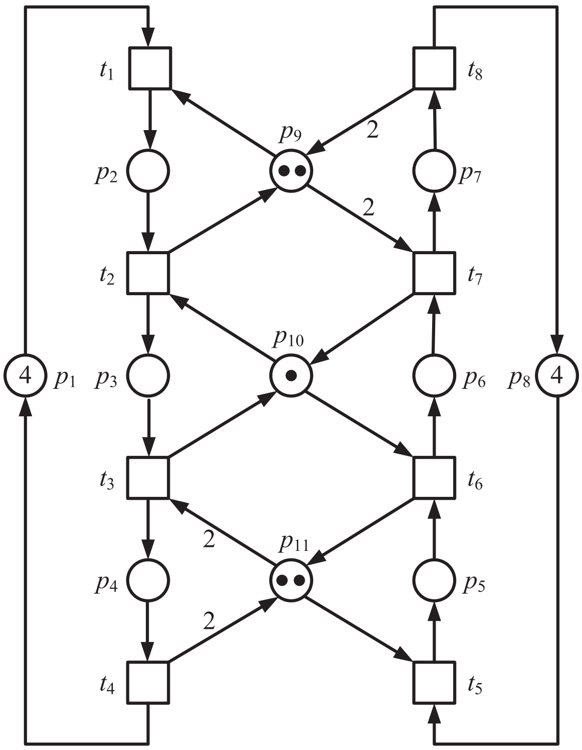

Fig. 1. Petri net model of an FMS.

Fig. 1 shows the Petri net model of an FMS. The places have the following set partitions: $P^0=\{p_1,p_8\}$, $P_R=\{p_9-p_{11}\}$, and $P_A=\{p_2-p_7\}$. The model has 39 reachable markings, 23 of which are legal. Its reachability graph is shown in Fig. 2. By using the marking reduction approach, we have $\mathcal{M}_L^\star=\{2p_5+p_6+p_7, 2p_2+p_3+p_4\}$. It has two critical transitions t_1 and t_5, i.e., $T_c=\{t_1,t_5\}$, and 10 MTSIs that are classified into two sets: $\Omega_{t_1}=\{(M_2,t_1), (M_6,t_1), (M_7,t_1), (M_{14},t_1), (M_{23},t_1)\}$ and $\Omega_{t_5}=\{(M_1,t_5), (M_3,t_5), (M_4,t_5), (M_8,t_5), (M_{16},t_5)\}$. By using the reduction technique, the two sets are reduced to be $\Omega_{t_1}^\star=\{(M_2,t_1),(M_7,t_1)\}$ and $\Omega_{t_5}^\star=\{(M_1,t_5),(M_4,t_5)\}$. Thus, the minimal covered set $\Omega^\star$ of MTSIs is $\Omega^\star=\{(M_2,t_1),(M_7,t_1),(M_1,t_5),(M_4,t_5)\}$. The set of t_1-enabled good markings is $\mathcal{E}_{t_1}=\{M_0,M_1,M_4,M_8,M_{10},M_{17},M_{27},M_{33}\}$ and the set of t_5-enabled good markings is $\mathcal{E}_{t_5}=\{M_0,M_2,M_7,M_{14},M_{24},M_{32},M_{36}\}$. By using the reduction technique, we obtain $\mathcal{E}_{t_1}^\star=\{M_{33}\}$ and $\mathcal{E}_{t_5}^\star=\{M_{36}\}$. Specially, we have $\mathcal{E}_{t_1}^\star=\{M_{33}=p_2+p_3+p_4\}$ and $\mathcal{E}_{t_5}^\star=\{M_{36}=\ p_5+p_6+p_7\}$.

First, we consider Algorithm 1. Suppose that there is a control place p_{s_1} with a self-loop associated with each critical transition, i.e., $W(p_{s_1},t_1)=W(t_1,p_{s_1})=w_1$ and $W(p_{s_1},t_5)=W(t_5,p_{s_1})=w_5$. No legal marking is forbidden by p_{s_1}. Therefore, considering the two legal markings in $\mathcal{M}_L^\star$, we have

$$2l_2+l_3+l_4\le\beta \quad \text{and}$$

$$2l_5+l_6+l_7\le\beta.$$

Control place p_{s_1} should not disable the firing of t_1 at its enabled good markings. By considering $\sum_{i=2}^{7} l_i\cdot[N](p_i,t_1)=l_2$, for

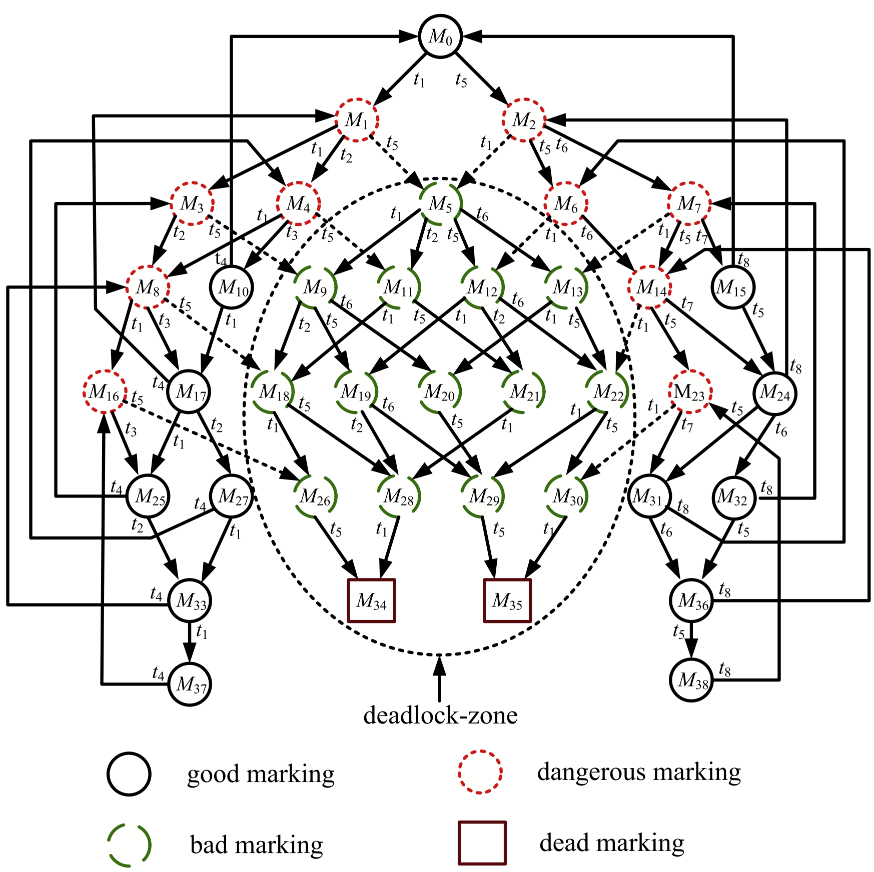

Fig. 2. The reachability graph of the net in Fig. 1.

Table 3
Experimental results for the net shown in Fig. 1 by Algorithm 1.

i	N_{con}	N_{var}	h^*	l_i	w_q	${}^\bullet p_{s_i}$	$p_{s_i}^\bullet$	$M_0(p_{s_i})$
1	8	13	2	$\mu_5+\mu_6\le 3$	$w_1=3$	$3t_1, t_7$	$3t_1, t_5$	3
2	5	10	2	$\mu_2+\mu_3\le 3$	$w_5=3$	$t_3, 3t_5$	$t_1, 3t_5$	3

the marking in $\mathcal{E}^\star_{t_1}$, we have

$$2l_2+l_3+l_4\le\beta-w_1.$$

Similarly, for t_5, we have $\sum_{i=2}^{7} l_i\cdot[N](p_i,t_5)=l_5$. Hence, the enabled condition for the marking in $\mathcal{E}^\star_{t_5}$ is:

$$2l_5+l_6+l_7\le\beta-w_5.$$

Four binary variables $f_{2,1}$, $f_{7,1}$, $f_{1,5}$, and $f_{4,5}$ are introduced to represent whether p_{s_1} implements the four MTSIs in $\Omega^\star$, i.e., (M_2,t_1), (M_7,t_1), (M_1,t_5), and (M_4,t_5), respectively. Hence, we have

$$l_2+l_5\ge -Q\cdot(1-f_{2,1})+\beta-w_1+1,$$
$$l_2+l_6\ge -Q\cdot(1-f_{7,1})+\beta-w_1+1,$$
$$l_2+l_5\ge -Q\cdot(1-f_{1,5})+\beta-w_5+1,\quad\text{and}$$
$$l_3+l_5\ge -Q\cdot(1-f_{4,5})+\beta-w_5+1.$$

An objective function is used to maximize the number of MTSIs that are implemented by p_{s_1}, as presented below:

$$\max f=f_{2,1}+f_{7,1}+f_{1,5}+f_{4,5}.$$

By summarizing the above constraints, we formulate an ILPP, IMNM, as follows:

IMNM:

$$\max f=f_{2,1}+f_{7,1}+f_{1,5}+f_{4,5}$$

subject to

$$2l_2+l_3+l_4\le\beta$$
$$2l_5+l_6+l_7\le\beta$$
$$2l_2+l_3+l_4\le\beta-w_1$$
$$2l_5+l_6+l_7\le\beta-w_5$$
$$l_2+l_5\ge -Q\cdot(1-f_{2,1})+\beta-w_1+1$$
$$l_2+l_6\ge -Q\cdot(1-f_{7,1})+\beta-w_1+1$$
$$l_2+l_5\ge -Q\cdot(1-f_{1,5})+\beta-w_5+1$$
$$l_3+l_5\ge -Q\cdot(1-f_{4,5})+\beta-w_5+1$$
$$l_2,l_3,l_4,l_5,l_6,l_7\in\{0,1,2,\ldots\}$$
$$\beta\in\{0,1,2,3,\ldots\}$$
$$w_1,w_5\in\{0,1,2,3,\ldots\}$$
$$f_{2,1},f_{7,1},f_{1,5},f_{4,5}\in\{0,1\}$$

The above ILPP has an optimal solution with $l_5=l_6=1$, $\beta=w_1=3$, and $f^*=2$. Since $w_5=0$ in the solution, there is no self-loop associated with t_5. Then, we can design a control place p_{s_1} with a self-loop associated with t_1, where p_{s_1} satisfies constraint: $\mu_5+\mu_6\le 3$ and the weight of the self-loop is 3. That is to say, ${}^\bullet p_{s_1}=\{3t_1,t_7\}$, $p_{s_1}^\bullet=\{3t_1,t_6\}$, and $M_0(p_{s_1})=3$. Removing the MTSIs implemented by p_{s_1}, we have $\Omega^\star=\{(M_1,t_5),(M_4,t_5)\}$.

At the second iteration, only t_5 is required to be disabled. Thus, we consider a control place p_{s_2} with a self-loop associated with t_5. Then, IMNM can be easily obtained by removing the enabled conditions for $\mathcal{E}^\star_{t_1}$ and the disabled conditions for MTSIs implemented by p_{s_1} from the IMNM at the first iteration, as presented below.

IMNM:

$$\max f=f_{1,5}+f_{4,5}$$

subject to

$$2l_2+l_3+l_4\le\beta$$
$$2l_5+l_6+l_7\le\beta$$

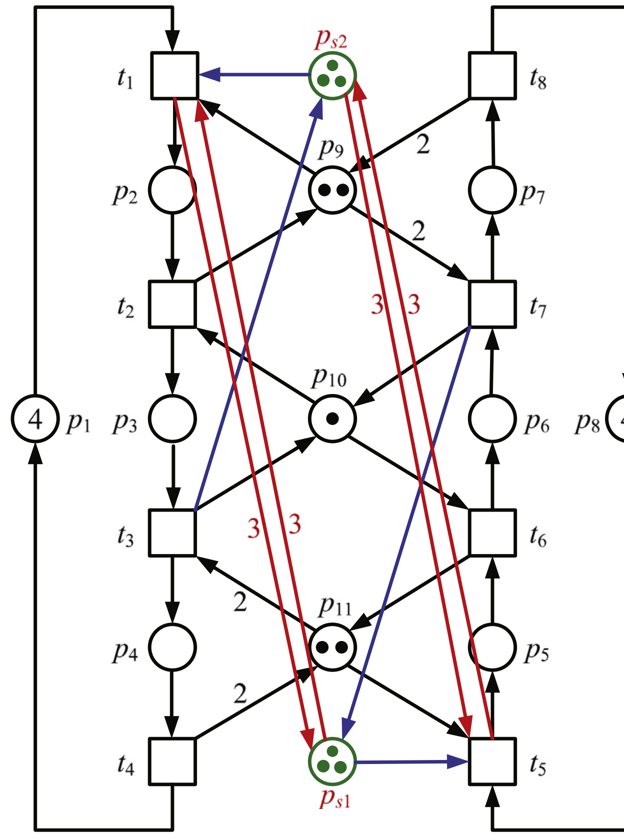

Fig. 3. Optimal Petri net supervisor for the net in Fig. 1.

$$2l_5+l_6+l_7\le\beta-w_5$$
$$l_2+l_5\ge -Q\cdot(1-f_{1,5})+\beta-w_5+1$$
$$l_3+l_5\ge -Q\cdot(1-f_{4,5})+\beta-w_5+1$$
$$l_2,l_3,l_4,l_5,l_6,l_7\in\{0,1,2,\ldots\}$$
$$\beta,w_5\in\{0,1,2,3,\ldots\}$$
$$f_{1,5},f_{4,5}\in\{0,1\}$$

The above ILPP has an optimal solution with $l_2=l_3=1$, $\beta=w_5=3$, and $f^*=2$. Then we can design a control place p_{s_2} with a self-loop associated with t_5, where p_{s_2} satisfies constraints: $\mu_2+\mu_3\le 3$ and hence the weight of the self-loop is 3. That is to say, ${}^\bullet p_{s_2}=\{t_3,3t_5\}$, $p_{s_2}^\bullet=\{t_1,3t_5\}$, and $M_0(p_{s_2})=3$.

Finally, two control places are obtained, which can make the net model live with all 23 legal markings. The results are shown in Table 3, where i is the iteration number, N_{con} and N_{var} indicate the numbers of constraints and variables in IMNM, respectively, l_i is the computed PI, w_q indicates the weight of the self-loop associated with t_q, and ${}^\bullet p_{s_i}$, $p_{s_i}^\bullet$, and $M_0(p_{s_i})$ in the seventh to ninth columns show the preset, poset, and initial marking of the computed control place p_{s_i}, respectively. The final optimally controlled system is shown in Fig. 3.

Next, we show MNCP for this example. Suppose that $n_c=|T_c|=2$, then MNCP is:

MNCP:

$$\min h=h_1+h_2$$

subject to

$$2l_{1,2}+l_{1,3}+l_{1,4}\le\beta_1 \tag{21}$$

$$2l_{1,5}+l_{1,6}+l_{1,7}\le\beta_1 \tag{22}$$

$$2l_{2,2}+l_{2,3}+l_{2,4}\le\beta_2 \tag{23}$$

$$2l_{2,5}+l_{2,6}+l_{2,7} \le \beta_2 \tag{24}$$

$$2l_{1,2}+l_{1,3}+l_{1,4} \le \beta_1 - w_{1,1} \tag{25}$$

$$2l_{1,5}+l_{1,6}+l_{1,7} \le \beta_1 - w_{1,5} \tag{26}$$

$$2l_{2,2}+l_{2,3}+l_{2,4} \le \beta_2 - w_{2,1} \tag{27}$$

$$2l_{2,5}+l_{2,6}+l_{2,7} \le \beta_2 - w_{2,5} \tag{28}$$

$$l_{1,2}+l_{1,5} \ge -Q\cdot(1-f_{1,2,1})+\beta_1-w_{1,1}+1 \tag{29}$$

$$l_{1,2}+l_{1,6} \ge -Q\cdot(1-f_{1,7,1})+\beta_1-w_{1,1}+1 \tag{30}$$

$$l_{1,2}+l_{1,5} \ge -Q\cdot(1-f_{1,1,5})+\beta_1-w_{1,5}+1 \tag{31}$$

$$l_{1,3}+l_{1,5} \ge -Q\cdot(1-f_{1,4,5})+\beta_1-w_{1,5}+1 \tag{32}$$

$$l_{2,2}+l_{2,5} \ge -Q\cdot(1-f_{2,2,1})+\beta_2-w_{2,1}+1 \tag{33}$$

$$l_{2,2}+l_{2,6} \ge -Q\cdot(1-f_{2,7,1})+\beta_2-w_{2,1}+1 \tag{34}$$

$$l_{2,2}+l_{2,5} \ge -Q\cdot(1-f_{2,1,5})+\beta_2-w_{2,5}+1 \tag{35}$$

$$l_{2,3}+l_{2,5} \ge -Q\cdot(1-f_{2,4,5})+\beta_2-w_{2,5}+1 \tag{36}$$

$$f_{1,2,1} \le h_1 \tag{37}$$

$$f_{1,7,1} \le h_1 \tag{38}$$

$$f_{1,1,5} \le h_1 \tag{39}$$

$$f_{1,4,5} \le h_1 \tag{40}$$

$$f_{2,2,1} \le h_2 \tag{41}$$

$$f_{2,7,1} \le h_2 \tag{42}$$

$$f_{2,1,5} \le h_2 \tag{43}$$

$$f_{2,4,5} \le h_2 \tag{44}$$

$$f_{1,2,1}+f_{2,2,1} \ge 1 \tag{45}$$

$$f_{1,7,1}+f_{2,7,1} \ge 1 \tag{46}$$

$$f_{1,1,5}+f_{2,1,5} \ge 1 \tag{47}$$

$$f_{1,4,5}+f_{2,4,5} \ge 1 \tag{48}$$

$$l_{1,2}, l_{1,3}, l_{1,4}, l_{1,5}, l_{1,6}, l_{1,7} \in \{0, 1, \ldots\}$$
$$l_{2,2}, l_{2,3}, l_{2,4}, l_{2,5}, l_{2,6}, l_{2,7} \in \{0, 1, \ldots\}$$
$$\beta_1, \beta_2 \in \{0, 1, \ldots\}$$
$$w_{1,1}, w_{1,5}, w_{2,1}, w_{2,5} \in \{0, 1,, \ldots\}$$
$$f_{1,2,1}, f_{1,7,1}, f_{1,1,5}, f_{1,4,5}, f_{2,2,1}, f_{2,7,1}, f_{2,1,5}, f_{2,4,5} \in \{0, 1\}$$

In MNCP, Eqs. (21)–(24) are reachability conditions, Eqs. (25)–(28) represent enabled conditions, and Eqs. (29)–(36) imply disabled conditions. Eqs. (37)–(44) indicate that a control place does not implement any MTSI if it is not selected. Eqs. (45)–(48) ensure that any MTSI must be implemented by at least one control place.

The above MNCP has 28 constraints and 26 variables. By solving it, we can obtain an optimal solution with $l_{1,5}=l_{1,6}=l_{2,2}=l_{2,3}=1$, $\beta_1=\beta_2=w_{1,1}=w_{2,5}=3$, and $h^*=2$. Thus, we can obtain two control places p_{s_1} and p_{s_2}. Specifically, we have ${}^\bullet p_{s_1}=\{3t_1, t_7\}$, $p_{s_1}^\bullet=\{3t_1, t_6\}$, $M_0(p_{s_1})=3$, ${}^\bullet p_{s_2}=\{t_3, 3t_5\}$, $p_{s_2}^\bullet=\{t_1, 3t_5\}$, and $M_0(p_{s_2})=3$. The results by MNCP are shown in Table 4, where z indicates the z-th control place, I_z is the z-th PI, $w_{z,q}$ indicates the weight of the self-loop between p_{s_z} and t_q, and ${}^\bullet p_{s_z}$, $p_{s_z}^\bullet$, and $M_0(p_{s_z})$ in the fourth to sixth columns show the preset, poset, and initial marking of the computed control place p_{s_z}, respectively. Note that MNCP obtains the same supervisor as Algorithm 1 does.

This net model has no optimal pure Petri net supervisor. Thus, we cannot find an optimal Petri net supervisor by the work in [25,52,6,7]. However, both the proposed approaches and the previous work [11] can provide a non-pure optimal supervisor with self-loops, which has the minimal number of control places. The final controlled system has all 23 legal markings. Thus, it can be seen that Petri net supervisors with self-loops are more powerful than pure net supervisors. In next section, some examples are

Table 4
Experimental results for the net shown in Fig. 1 by MNCP.

z	I_z	$w_{z,q}$	${}^\bullet p_{s_z}$	$p_{s_z}^\bullet$	$M_0(p_{s_z})$
1	$\mu_5+\mu_6 \le 3$	$w_{1,1}=3$	$3t_1, t_7$	$3t_1, t_5$	3
2	$\mu_2+\mu_3 \le 3$	$w_{2,5}=3$	$t_3, 3t_5$	$t_1, 3t_5$	3

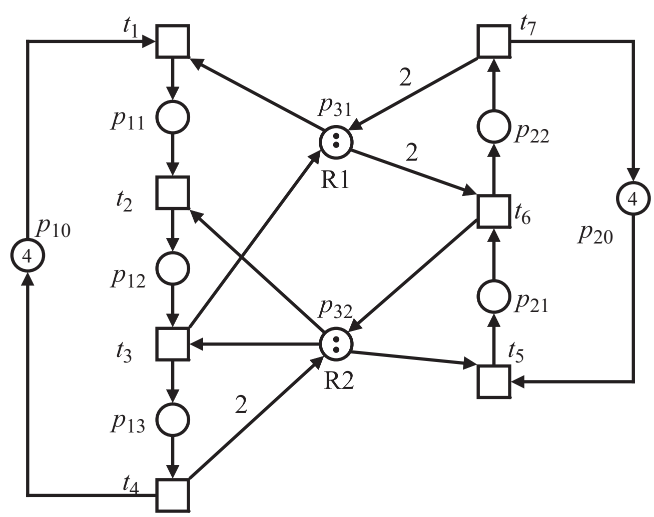

Fig. 4. A Petri net model in Tricas et al. [51].

Table 5
Experimental results for the net shown in Fig. 4 by Algorithm 1.

i	N_{con}	N_{var}	h^*	I_i	w_q	${}^\bullet p_{s_i}$	$p_{s_i}^\bullet$	$M_0(p_{s_i})$
1	11	13	3	$\mu_{11}+2\mu_{12} \le 3$	$w_5=3$	$2t_3, 3t_5$	$t_1, t_2, 3t_5$	3
2	6	8	1	$\mu_{21} \le 2$	$w_1=2$	$2t_1, t_6$	$2t_1, t_5$	2

Table 6
Experimental results for the net shown in Fig. 4 by MNCP.

z	I_z	$w_{z,q}$	${}^\bullet p_{s_z}$	$p_{s_z}^\bullet$	$M_0(p_{s_z})$
1	$\mu_{21} \le 2$	$w_{1,1}=2$	$2t_1, t_6$	$2t_1, t_5$	2
2	$\mu_{11}+2\mu_{21} \le 3$	$w_{2,5}=3$	$2t_3, 3t_5$	$t_1, t_2, 3t_5$	3

Table 7
Performance comparison of deadlock control policies for the S[4]PR in Fig. 4.

Parameters	[51]	[10]	[11]	Alg. 1	MNCP
No. monitors	4	3	3	2	2
No. arcs	13	10	10	9	9
No. markings	10	11	13	13	13

provided to show that both Algorithm 1 and MNCP can find an optimal supervisor with less number of control places than that proposed in the previous work [11].

4. Experimental results

In this section, two Petri net models of FMSs are used to illustrate the proposed methods. First, we consider an S^{4}PR from [51], as shown in Fig. 4. There are nine places and seven transitions and the places have the following set partitions: $P^0=\{p_{10},p_{20}\}$, $P_R=\{p_{31},p_{32}\}$, and $P_A=\{p_{11},p_{12},p_{13},p_{21},p_{22}\}$. It has 20 reachable markings, 13 of which are legal. By using the vector covering approach, $\mathcal{M}_L^{\star}$ has three markings only, i.e., $\mathcal{M}_L^{\star}=\{p_{11}+p_{12},2p_{21}+p_{22},2p_{11}+p_{13}\}$. The net has seven MTSIs and three critical transitions t_1, t_2, and t_5, where $|\mathcal{E}_{t_1}|=5$, $|\mathcal{E}_{t_2}|=2$, and $|\mathcal{E}_{t_5}|=4$. By using the marking reduction approach, we have $|\mathcal{E}^{\star}_{t_1}|=2$, $|\mathcal{E}^{\star}_{t_2}|=1$, $|\mathcal{E}^{\star}_{t_5}|=1$, and $|\Omega^{\star}|=4$. Tables 5 and 6 show the applications of Algorithm 1 and MNCP, respectively. For this net model, MNCP with $n_c=2$ has 34 constraints and 28 variables.

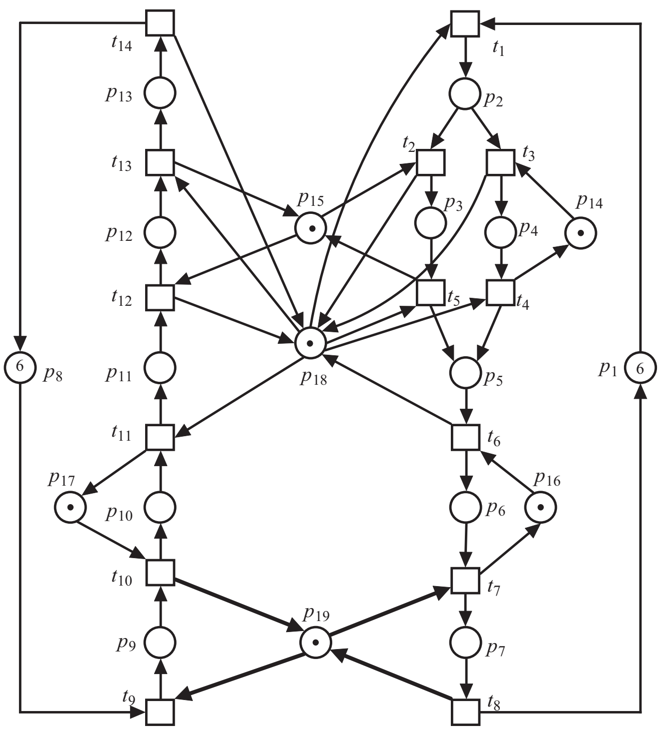

Fig. 5. Petri net model of an FMS.

The net model does not have an optimal pure Petri net supervisor by using the theory of regions in [25,52] or the methods proposed in the previous work [6,7]. In the previous work [12], we propose an optimal supervisor to enforce disjunctive constraints. However, the supervisory structure in [12] includes four added transitions, which lead to six extra markings in the controlled system. The previous work [11] and the proposed methods in this paper can find optimal supervisors with

Table 10
Performance comparison of some deadlock control policies for the net shown in Fig. 5.

Parameters	[52]	[35]	[46]	[6]	[7]	[11]	[66]	[14]	Alg. 1	MNCP
No. monitors	6	9	5	8	2	5	2	2	2	2
No. arcs	32	42	23	37	12	40	12	12	16	14
No. markings	205	205	205	205	205	205	205	205	205	205

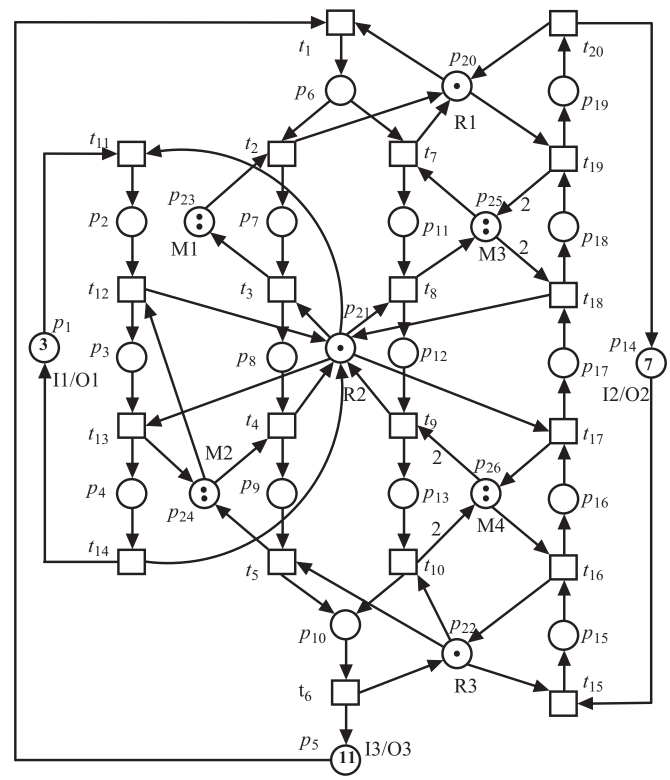

Fig. 6. Petri net model of an FMS.

Table 8
Experimental results for the net shown in Fig. 5 by Algorithm 1.

i	N_{con}	N_{var}	h^*	I_i	w_q	${}^\bullet p_{s_i}$	$p_{s_i}^\bullet$	$M_0(p_{s_i})$
1	71	29	8	$\mu_2+2\mu_3+\mu_4+2\mu_5+2\mu_6+3\mu_9+3\mu_{10}\le 9$	$w_{11}=3$	$2t_7,3t_{11}$	$t_1,t_2,t_4,3t_9$	9
2	40	18	4	$2\mu_3+\mu_4+2\mu_{12}\le 3$	$w_1=1$, $w_{11}=2$	$t_1,t_4,2t_5,2t_{11},2t_{13}$	$t_1,2t_2,t_3,2t_{11},2t_{12}$	3

Table 9
Experimental results for the net shown in Fig. 5 by MNCP.

z	I_z	$w_{z,q}$	${}^\bullet p_{s_z}$	$p_{s_z}^\bullet$	$M_0(p_{s_z})$
1	$\mu_2+2\mu_3+\mu_4+2\mu_5+2\mu_6+3\mu_9+3\mu_{10}\le 9$	$w_{1,q}=0, q\in\{1,2,4,9,11\}$	$2t_7,3t_{11}$	$t_1,t_2,t_4,3t_9$	9
2	$\mu_2+2\mu_3+\mu_4+2\mu_{12}\le 3$	$w_{2,11}=2$	$t_4,2t_5,2t_{11},2t_{13}$	$t_1,t_2,2t_{11},2t_{12}$	3

Table 11
Experimental results for the net shown in Fig. 6 by Algorithm 1.

i	N_{con}	N_{var}	h^*	I_i	w_q	$^\bullet p_{s_i}$	$p_{s_i}^\bullet$	$M_0(p_{s_i})$
1	47	260	18	$8\mu_2+8\mu_3+2\mu_6+2\mu_7+9\mu_8+9\mu_9+45\mu_{11}+45\mu_{12}+45\mu_{13}+160\mu_{15}+10\mu_{16}+\mu_{17}+\mu_{18}\le 204$	$w_3=1$, $w_7=2$, $w_{17}=9$	$t_3, 9t_5, 2t_7, 45t_{10}$, $8t_{13}, 150t_{16}, 9t_{17}$, t_{19}	$2t_1, 8t_3, 45t_7, 8t_{11}$, $160t_{15}$	204
2	204	29	3	$\mu_3+3\mu_{17}\le 5$	$w_3=4$, $w_7=3$, $w_{11}=4$	$4t_3, 3t_7, 4t_{11}, t_{13}$, $3t_{18}$	$4t_3, 3t_7, 4t_{11}, t_{12}$, $3t_{17}$	5
3	133	23	2	$\mu_6+\mu_7+\mu_{17}+\mu_{18}\le 4$	$w_7=2$	$t_3, 2t_7, t_{19}$	t_1, t_7, t_{17}	4
4	97	19	1	$\mu_{16}\le 2$	$w_7=2$	$2t_7, t_{17}$	$2t_7, t_{16}$	2

Table 12
Experimental results for the net shown in Fig. 6 by MNCP.

z	I_z	$w_{z,q}$	$^\bullet p_{s_z}$	$p_{s_z}^\bullet$	$M_0(p_{s_z})$
1	$\mu_2+\mu_3+\mu_8+\mu_9+5\mu_{11}+\mu_{12}+5\mu_{13}+14\mu_{15}+\mu_{16}\le 18$	$w_{1,q}=0,\ q\in\{1,3,5,7,11,17\}$	$t_5, 4t_8, 5t_{10}, t_{13}, 13t_{16}, t_{17}$	$t_3, 5t_7, 4t_9, t_{11}, 14t_{15}$	9
2	$2\mu_6+2\mu_7+\mu_8+\mu_9+\mu_{12}+2\mu_{15}+8\mu_{16}+\mu_{17}+\mu_{18}\le 26$	$w_{2,3}=1, w_{2,7}=19$	$t_3, t_5, 21t_7, t_9, 7t_{17}, t_{19}$	$2t_1, 19t_7, t_8, 2t_{15}, 6t_{16}$	26
3	$\mu_6+\mu_7+3\mu_{17}+\mu_{18}\le 6$	$w_{3,7}=4$	$t_3, 4t_7, 2t_{18}, t_{19}$	$t_1, 3t_7, 3t_{17}$	6
4	$\mu_2+\mu_3+\mu_8+3\mu_{12}\le 5$	$w_{4,3}=w_{4,11}=w_{4,15}=3$	$3t_3, t_4, 3t_9, 3t_{11}, t_{13}, 3t_{15}$	$4t_3, 3t_8, 4t_{11}, 3t_{15}$	5

self-loops. Table 7 shows the performance comparison of some policies for this example. The work in [11] and the proposed approaches can lead to optimal supervisors to make all 13 legal markings reachable. Note that in [11], the optimal supervisor has three control places and 10 arcs. However, considering our solutions to this problem, both Algorithm 1 and MNCP can obtain an optimal supervisor with only two control places and nine arcs, which is the simplest supervisory structure.

Next, we consider a well-known example as shown in Fig. 5. It is studied in several papers (see [35,46,47,52]). There are 19 places and 14 transitions. The places have the following set partitions: $P^0=\{p_1, p_8\}$, $P_R=\{p_{14}-p_{19}\}$, and $P_A=\{p_2-p_7, p_9-p_{13}\}$. It has 282 reachable markings, 205 of which are legal ones. By using a vector covering approach, $\mathcal{M}_L^\star$ has only 26 markings. The net has 59 MTSIs and five critical transitions t_1, t_2, t_4, t_9, and t_{11}, where $|\mathcal{E}_{t_1}|=46$, $|\mathcal{E}_{t_2}|=22$, $|\mathcal{E}_{t_4}|=32$, $|\mathcal{E}_{t_9}|=63$, and $|\mathcal{E}_{t_{11}}|=12$. By using the marking reduction approach, we have $|\mathcal{E}^\star_{t_1}|=8$, $|\mathcal{E}^\star_{t_2}|=3$, $|\mathcal{E}^\star_{t_4}|=5$, $|\mathcal{E}^\star_{t_9}|=15$, $|\mathcal{E}^\star_{t_{11}}|=2$, and $|\Omega^\star|=12$. Tables 8 and 9 show the applications of Algorithm 1 and MNCP, respectively. For this net model, MNCP with n_c=2 has 178 constraints and 60 variables. Note that in the first row of Table 9, we obtain $w_{1,q}=0$, $\forall t_q\in T_c$, i.e., no self-loop is added for control place p_{s_1}.

Table 10 shows the performance comparison of some policies for this example. It can be seen that the previous work [11] obtains an optimal solution with $|T_c|=5$ control places, which is bounded by $|T|$. However, both Algorithm 1 and MNCP can find an optimal supervisor with only two control places, which is three less than that in the previous work [11] and the same as the minimal structure proposed in [66]. However, the proposed methods lead to more arcs in the obtained supervisors. Note that in [14], we propose an approach to find an optimal supervisor with the lowest implementation cost, which also leads to the minimal supervisory structure with two control places and 12 arcs.

Consider another Petri net model of an FMS, as shown in Fig. 6. It is a WS^{3}PR that is a weighted version of an S^{3}PR in [21]. Comparing with S^{3}PR, both operation places p_{13} and p_{18} in this model have weighted resource requirements. There are 26 places and 20 transitions and the places have the following set partitions: $P^0=\{p_1, p_5, p_{14}\}$, $P_R=\{p_{20}-p_{26}\}$, and $P_A=\{p_2-p_4, p_6-p_{13}, p_{15}-p_{19}\}$. It has 9572 reachable markings, 5395 of which are legal ones. By using a vector covering approach, $\mathcal{M}_L^\star$ has only 87 markings. The net has 1915 MTSIs and six critical transitions t_1, t_3, t_7, t_{11}, t_{15}, and t_{17}, where $|\mathcal{E}_{t_1}|=2040$, $|\mathcal{E}_{t_3}|=715$, $|\mathcal{E}_{t_7}|=600$, $|\mathcal{E}_{t_{11}}|=1073$, $|\mathcal{E}_{t_{15}}|=1244$, and $|\mathcal{E}_{t_{17}}|=611$. By using the marking reduction approach, we have $|\mathcal{E}^\star_{t_1}|=47$, $|\mathcal{E}^\star_{t_3}|=17$, $|\mathcal{E}^\star_{t_7}|=9$, $|\mathcal{E}^\star_{t_{11}}|=17$, $|\mathcal{E}^\star_{t_{15}}|=38$, $|\mathcal{E}^\star_{t_{17}}|=21$, and $|\Omega^\star|=24$. Tables 11 and 12 show the applications of Algorithm 1 and MNCP, respectively. For this net model, MNCP with n_c=4 has 1352 constraints and 192 variables. Note that in the first row of Table 12, we obtain $w_{1,q}=0$, $\forall t_q\in T_c$, i.e., no self-loop is added for control place p_{s_1}.

Table 13
Performance comparison of some deadlock control policies for the net shown in Fig. 6.

Parameters	[10]	[11]	Alg. 1	MNCP
No. monitors	12	8	4	4
No. arcs	57	42	33	39
No. markings	4795	5395	5395	5395

Table 13 shows the performance comparison of some policies for this example. It can be seen that the previous work [11] obtains an optimal Petri net supervisor with eight control places and 42 arcs. However, both Algorithm 1 and MNCP can find optimal supervisors with only four control places and 33 and 39 arcs, respectively, which are four less control places and less arcs than that in the previous work [11], i.e., it is the minimal supervisory structure (Table 13).

5. Conclusions

This paper designs optimal nonpure Petri net supervisors for FMSs by improving the work in [11]. Self-loops are allowed in the supervisor to prevent the illegal markings. Two ILPPs are developed, namely IMNM and MNCP. The first, IMNM, designs a control place to implement as many MTSIs as possible. Thus, we can obtain an optimal supervisor with a small number of control places. The second ILPP, MNCP, can compute all control places at a time and its objective function is used to select the minimal number of control places. Compared with the work in [6,7,25,52,66], the advantage of the proposed methods is that the obtained supervisors with self-loops have more control ability than pure Petri nets. The experimental results show that the proposed work can find optimal Petri net supervisors for the net models that cannot be optimally controlled by pure Petri net supervisors. Compared with the work in [12], the obtained supervisor in this work does not include added transitions and no extra markings generated in the controlled system. Compared with the previous work [11], this research can obtain optimal supervisors with less control places and MNCP can even find the minimal number of control places.

Though this work can well handle the behavioral permissiveness and structural complexity problems, it suffers from the computational complexity problem. It requires to generate the reachability graph of a Petri net model and solve ILPPs that are NP-hard in theory. Although we use a marking reduction technique to reduce the numbers of constraints and variables in the ILPPs, which can greatly reduce the computational burden of the proposed methods, their computational complexity is still NP-hard. In the future, we will focus on avoiding the full enumeration of reachable markings [43] and reduce the number of constraints in the proposed ILPPs.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant nos. 61203038, 61374068, 61304050, 61304051, and 61364004, the Fundamental Research Funds for the Central Universities under Grant no. JB140402, the Recruitment Program of Global Experts, the Science and Technology Development Fund, MSAR, under Grant no. 066/2013/A2, and the Scientific and Technological Research Council of Turkey (Türkiye Bilimsel ve Teknolojik Araştırma Kurumu - TÜBİTAK) under Grant no. TÜBİTAK-112M229.

References

[1] Banaszak Z, Krogh BH. Deadlock avoidance in flexible manufacturing systems with concurrently competing process flows. IEEE Trans Robot Autom 1990;6(6):724–34.
[2] Barkaoui K, Abdallah IB. A deadlock prevention method for a class of FMS. In: Proceedings of IEEE international conference on systems, man and cybernetics, intelligent systems for the 21st century, Vancouver, BC; 1995. p. 4119–24.
[3] Barkaoui K, Chaoui A, Zouari B. Supervisory control of discrete event systems based on structure theory of Petri nets. In: Proceedings of IEEE international conference on systems, man, and cybernetics, computational cybernetics and simulation, vol. 4, Orlando, FL; 1997. p. 3750–5.
[4] Basile F, Chiacchio P. On the implementation of supervised control of discrete event systems. IEEE Trans Control Syst Technol 2007;15(4):725–39.
[5] Chao DY. Improvement of suboptimal siphon- and FBM-based control model of a well-known S^3PR. IEEE Trans Autom Sci Eng 2011;8(2):404–11.
[6] Chen YF, Li ZW, Khalgui M, Mosbahi O. Design of a maximally permissive liveness-enforcing Petri net supervisor for flexible manufacturing systems. IEEE Trans Autom Sci Eng 2011;8(2):374–93.
[7] Chen YF, Li ZW. Design of a maximally permissive liveness-enforcing supervisor with a compressed supervisory structure for flexible manufacturing systems. Automatica 2011;47(5):1028–34.
[8] Chen YF, Li ZW. On structural minimality of optimal supervisors for flexible manufacturing systems. Automatica 2012;48(10):2647–56.
[9] Chen YF, Li ZW, Zhou MC. Behaviorally optimal and structurally simple liveness-enforcing supervisors of flexible manufacturing systems. IEEE Trans Syst Man Cybern Part A: Syst Hum 2012;42(3):615–29.
[10] Chen YF, Li ZW, Zhou MC. Most permissive Petri net supervisors for flexible manufacturing systems. Int J Prod Res 2012;50(22):6357–71.
[11] Chen YF, Li ZW, Al-Ahmari A. Nonpure Petri net supervisors for optimal deadlock control of flexible manufacturing systems. IEEE Trans Syst Man Cybern Syst 2013;43(2):252–65.
[12] Chen YF, Li ZW, Zhou MC. Optimal supervisory control of flexible manufacturing systems by Petri nets: a set classification approach. IEEE Trans Autom Sci Eng 2014;11(2):549–63.
[13] Chen YF, Li ZW, Barkaoui K, Uzam M. New Petri net structure and its application to optimal supervisory control: interval inhibitor arcs. IEEE Trans Syst Man Cybern Syst 2014;44(10):1384–400.
[14] Chen YF, Li ZW, Barkaoui K. Maximally permissive liveness-enforcing supervisor with lowest implementation cost for flexible manufacturing systems. Inf Sci 2014;256:74–90.
[15] Chen YF, Li ZW, Barkaoui K, Giua A. On the enforcement of a class of nonlinear constraints on Petri nets. Automatica 2015;55(5):116–24.
[16] Coffman EG, Elphick MJ, Shoshani A. Systems deadlocks. ACM Comput Surv 1971;3(2):67–78.
[17] Cordone R, Piroddi L. Parsimonious monitor control of petri net models of FMS. IEEE Trans Syst Man Cybern Syst 2013;43(1):215–21.
[18] Das SK, Agrawal VK, Sarkar D, Patnaik LM, Goel PS. Reflexive incidence matrix (RIM) representation of Petri nets. IEEE Trans Softw Eng 1987;SE-13(6):643–53.
[19] Dideban A, Zareiee M, Alla H. Controller synthesis with highly simplified linear constraints in PN model. Asian J Control 2013;15(1):80–94.
[20] Dideban A, Zareiee M, Orouji AA. A simple Petri net controller by solving some integer linear programming problem. J Control Eng Appl Inf 2014;16(4):3–11.
[21] Ezpeleta J, Colom JM, Martinez JA. Petri net based deadlock prevention policy for flexible manufacturing systems. IEEE Trans Robot Autom 1995;11(2):173–84.
[22] Ezpeleta J, Tricas F, Garcia-Valles F, Colom JM. A banker's solution for deadlock avoidance in FMS with flexible routing and multiresource states. IEEE Trans Robot Autom 2002;18(4):621–5.
[23] Fanti MP, Zhou MC. Deadlock control methods in automated manufacturing systems. IEEE Trans Syst Man Cybern Part A 2004;34(1):5–22.
[24] Ferrarini L, Piroddi L. Modeling and control of fluid transportation operations in production plants with Petri nets. IEEE Trans Control Syst Technol 2008;16(5):1090–8.
[25] Ghaffari A, Rezg N, Xie XL. Design of a live and maximally permissive Petri net controller using the theory of regions. IEEE Trans Robot Autom 2003;19(1):137–42.
[26] Hsieh FS, Chang SC. Dispatching-driven deadlock avoidance controller synthesis for flexible manufacturing systems. IEEE Trans Robot Autom 1994;10(2):196–209.
[27] Hu HS, Zhou MC, Li ZW. Algebraic synthesis of timed supervisor for automated manufacturing systems using Petri nets. IEEE Trans Autom Sci Eng 2010;7(3):549–57.
[28] Huang YS, Jeng MD, Xie XL, Chung DH. Siphon-based deadlock prevention for flexible manufacturing systems. IEEE Trans Syst Man Cybern Part A 2006;36(6):1248–56.
[29] Li ZW, Zhou MC. Elementary siphons of Petri nets and their application to deadlock prevention in flexible manufacturing systems. IEEE Trans Syst Man Cybern Part A 2004;34(1):38–51.
[30] Li ZW, Zhou MC. Clarifications on the definitions of elementary siphons of Petri nets. IEEE Trans Syst Man Cybern Part A 2006;36(6):1227–9.
[31] Li ZW, Zhou MC. Two-stage method for synthesizing liveness-enforcing supervisors for flexible manufacturing systems using Petri nets. IEEE Trans Ind Informatics 2006;2(4):313–25.
[32] Li ZW, Hu HS, Wang AR. Design of liveness-enforcing supervisors for flexible manufacturing systems using Petri nets. IEEE Trans Syst Man Cybern Part C 2007;37(4):517–26.
[33] Li ZW, Zhao M. On controllability of dependent siphons for deadlock prevention in generalized Petri nets. IEEE Trans Syst Man Cybern Part A 2008;38(2):369–84.
[34] Li ZW, Zhou MC. Control of elementary and dependent siphons in Petri nets and their application. IEEE Trans Syst Man Cybern Part A 2008;38(1):133–48.
[35] Li ZW, Zhou MC, Jeng MD. A maximally permissive deadlock prevention policy for FMS based on Petri net siphon control and the theory of regions. IEEE Trans Autom Sci Eng 2008;5(1):182–8.
[36] Li ZW, Zhou MC, Wu NQ. A survey and comparison of Petri net-based deadlock prevention policies for flexible manufacturing systems. IEEE Trans Syst Man Cybern Part C 2008;38(2):173–88.
[37] Li ZW, Liu GY, Hanisch MH, Zhou MC. Deadlock prevention based on structure reuse of Petri net supervisors for flexible manufacturing systems. IEEE Trans Syst Man Cybern Part A 2012;42(1):178–91.
[38] Li ZW, Wu NQ, Zhou MC. Deadlock control of automated manufacturing systems based on Petri nets—a literature review. IEEE Trans Syst Man Cybern Part C 2012;42(4):437–62.
[39] Liu D, Li ZW, Zhou MC. Liveness of an extended S^3PR. Automatica 2010;46(6):1008–18.
[40] Ma ZY, Li ZW, Giua A. Design of optimal Petri net controllers for disjunctive generalized mutual exclusion constraints. IEEE Trans Autom Control 2015;60(7):1774–85.
[41] Moody JO, Antsaklis PJ. Supervisory control of discrete event systems using Petri nets. Amsterdam, The Netherlands: Kluwer Academic Publishers; 1998.
[42] Murata T. Petri nets: properties, analysis and application. Proc IEEE 1989;77(4):541–80.
[43] Nazeem A, Reveliotis S. An efficient algorithm for the enumeration of the minimal unsafe states in complex resource allocation systems. In: Proceedings of the IEEE international conference on automation science and engineering, Seoul, Korea; 2012. p.686–93.
[44] Park J, Reveliotis SA. Algebraic synthesis of efficient deadlock avoidance policies for sequential resource allocation systems. IEEE Trans Robot Autom 2000;16(2):190–5.
[45] Park J, Reveliotis SA. Deadlock avoidance in sequential resource allocation systems with multiple resource acquisitions and flexible routings. IEEE Trans Autom Control 2001;46(10):1572–83.
[46] Piroddi L, Cordone R, Fumagalli I. Selective siphon control for deadlock prevention in Petri nets. IEEE Trans Syst Man Cybern Part A 2008;38(6):1337–48.

[47] Piroddi L, Cordone R, Fumagalli I. Combined siphon and marking generation for deadlock prevention in Petri nets. IEEE Trans Syst Man Cybern Part A 2009;39(3):650–61.
[48] Tarek A, Lopez-Benitez N. Optimal legal firing sequence of Petri nets using linear programming. Optim Eng 2004;5(1):25–43.
[49] Tricas F, Garcia-Valles F, Colom JM, Ezpeleta JA. structural approach to the problem of deadlock prevention in processes with resources. In: Proceedings of WODES'98, Italy; 1998. p. 273–8.
[50] Tricas F, Garcia-Valles F, Colom JM, Ezpeleta JA. An iterative method for deadlock prevention in FMS. In: Stremersch G, editor. Discrete event systems: analysis and control. Kluwer Academic: Boston, MA; 2000. p. 139–48.
[51] Tricas F, Garcia-Valles F, Colom JM, Ezpeleta JA. A Petri net structure-based deadlock prevention solution for sequential resource allocation systems. In: Proceedings of the IEEE International Conference on Robotics and Automation, Barcelona, Spain; 2005. p. 271–7.
[52] Uzam M. An optimal deadlock prevention policy for flexible manufacturing systems using Petri net models with resources and the theory of regions. Int J Adv Manuf Technol 2002;19(3):192–208.
[53] Uzam M, Zhou MC. An improved iterative synthesis method for liveness enforcing supervisors of flexible manufacturing systems. Int J Prod Res 2006;44(10):1987–2030.
[54] Uzam M, Zhou MC. An iterative synthesis approach to Petri net based deadlock prevention policy for flexible manufacturing systems. IEEE Trans Sys Man Cybern Part A 2007;37(3):362–71.
[55] Wang HQ, Jiang CJ, Liao SY. Behavior relations in synthesis process of Petri net models. IEEE Trans Robot Autom 2000;16(4):400–7.
[56] Wikipedia. Flexible manufacturing systems, ⟨http://en.wikipedia.org/wiki/Flexible_manufacturing_system⟩; 2015.
[57] Wu NQ, Zhou MC. Deadlock resolution in automated manufacturing systems with robots. IEEE Trans Autom Sci Eng 2007;4(3):474–80.
[58] Wu NQ, Zhou MC, Li ZW. Resource-oriented Petri net for deadlock avoidance in flexible assembly systems. IEEE Trans Syst Man Cybern Part A 2008;38(1):56–69.
[59] Wu NQ, Zhou MC, Li ZW. Short-term scheduling of crude-oil operations: petri net-based control-theoretic approach. IEEE Robot Autom Mag 2015;22(2):64–76.
[60] Xing KY, Hu BS, Chen HX. Deadlock avoidance policy for flexible manufacturing systems. In: Zhou MC, editor. Petri nets in flexible and agile automation. Boston, MA: Kluwer Academic; 1995. p. 239–63.
[61] Xing KY, Zhou MC, Wang F, Liu HX, Tian F. Resource-transition circuits and siphons for deadlock control of automated manufacturing systems. IEEE Trans Syst Man Cybern Part A 2011;41(1):74–84.
[62] Yamalidou K, Moody J, Lemmon M, Antsaklis P. Feedback control of Petri nets based on place invariants. Automatica 1996;32(1):15–28.
[63] Ye JH, Li ZW, Giua A. Decentralized supervision of Petri nets with a coordinator. IEEE Trans Syst Man Cybern Syst 2015;45(6):955–66.
[64] Zareiee M, Dideban A, Nazemzadeh P. From forbidden states to linear constraints. Dubai, UME: World Academy of Science, Engineering and Technology; 2011.
[65] Zareiee M, Dideban A, Orouji AA, Soleymanpour HR. Simplification of a Petri net controller in industrial systems by using an optimization algorithm. Int J Ind Eng Prod Res 2013;24(1):59–70.
[66] Zareiee M, Dideban A, Orouji AA. Safety analysis of discrete event systems using a simplified Petri net controller. ISA Trans 2014;53(1):44–9.
[67] Zareiee M, Dideban A, Orouji AA, Soltanizadeh H. Solving the problem of forbidden states in discrete event systems: a novel systematic method for reducing the number of control places. Asian J Control 2015;17(3):1006–15.
[68] Zhong CF, Li ZW, Chen YF, Al-Ahmari A. On nonexistence of a maximally permissive liveness-enforcing pure net supervisor. IEEE Trans Syst Man Cybern Syst 2013;43(1):29–37.
[69] Zhou MC, Wang CH, Zhao XY. Automating Mason's rule and its application to analysis of stochastic Petri nets. IEEE Trans Control Syst Technol 1995;3(2):238–44.