CEAI, Vol.23, No. 4, pp. 46-56, 2021 Printed in Romania

A Think-Globally-Act-Locally-Based Method of Maximally Permissive Liveness-Enforcing Supervisors for Flexible Manufacturing Systems ⋆

Chengzong Li * **Yufeng Chen** * **Zhaofeng Zhong** ** **Murat Uzam** *** **Zhiwu Li** * **Naiqi Wu** * **Mingjun Zhang** ****

* *Institute of Systems Engineering, Macau University of Science and Technology, Taipa, Macau 999078, China (e-mail: lczxdu@163.com, Corresponding author: chyf01@163.com, systemscontrol@gmail.com, nqwu@must.edu.mo)*
** *Hitachi Building Technology (Guangzhou) Co.,Ltd, Guangzhou 510330, China*
*** *Elektrik-Elektronik Muhendisligi Bolumu, Muhendislik-Mimarlik Fakultesi, Yozgat Bozok Universitesi, Yozgat 66100, Turkey (e-mail: murat_uzam@hotmail.com)*
**** *Dgut-Cnam Institute, Dongguan University of Technology, Dongguan 523808, China (e-mail: ming-jun.zhang@lecnam.net)*

Abstract: With a vector covering (VC) technique, this study proposes a think-globally-act-locally-based (TGAL for short) approach in order to design optimal supervisors for flexible manufacturing systems (FMSs). First, a global idle place is added into a Petri net (PN) model of an FMS according to the TGAL concept. With the global idle place being marked by one token at the initial marking, the model's reachability graph is generated to analyze its liveness. If it is not live, one finds first-met bad markings (FBMs) and all legal markings. By applying the VC approach, these markings can be covered by a minimal covered set (MCdS) of FBMs and a minimal covering set (MCgS) of legal markings, respectively, to remarkably reduce the number of markings under consideration. For each FBM in the MCdS of FBMs, we modify the marking in the MCgS of FBM-related legal markings to guarantee that every place has the maximal tokens. By using the modified FBM-related legal markings, a set of optimal control places (monitors) can be calculated by solving an integer linear programming problem. Necessary monitors are figured out by a redundancy test. Then, we increase the number of tokens in the global idle place. This process is repeated until all legal markings are reachable. Then, a maximally permissive supervisor can be obtained. Finally, by means of experimental studies, the proposed method is tested for some PN models, showing that the designed supervisors are maximally permissive since the monitors do not prohibit any legal markings.

Keywords: Flexible manufacturing system, Petri net, think-globally-act-locally approach, supervisory control, deadlock prevention, optimal liveness-enforcing supervisor.

1. INTRODUCTION

Flexible manufacturing systems (FMSs) are deployed in both process and discrete manufacturing industries. In an FMS, multiple processes for different products share the versatile resources. Deadlocks would occur, if resources are not properly allocated. In a deadlock state, a circular wait is formed such that some processes wait for the availability of particular resources that can never be released (Coffman et al., 1971). Deadlocks usually lead to a system blockage with a low efficiency. Thus, deadlocks are usually thought of as an unacceptable status for FMSs. In order to prevent deadlocks, there are a number of control policies reported in the literature (Basile et al., 2015; Chen and Li, 2012; Chen et al., 2015; Cordone et al., 2013; Hu and Zhou, 2014; Huang et al., 2006; Ma et al., 2015; Wang et al., 2015; Wu and Zhou, 2012; Xing et al., 2010).

As a useful tool for deadlock detection and resolution, PNs are employed to model, analyze, and control FMSs (Ezpeleta et al., 1995; Li et al., 2007; Li and Zhou, 2006b; Zhu et al., 2017, 2018; Zhang et al., 2017). For a PN model, deadlocks are prevented by adding a set of control places (monitors) to form a supervisor by enforcing some necessary constraints on the system. The advantage of deadlock prevention is that one can compute a supervisor in an off-line way. Deadlock prevention policies have been extensively developed by using PN models (Li and Zhou,

⋆ This work was supported by the FDCT, MSAR SAR, under Grants 011/2017/A, 0012/2019/A1, and 0064/2021/A2, and Guangzhou Innovation and Entrepreneurship Leading Team Project Funding under Grant NO.202009020008.

2004, 2006a; Chen et al., 2014, 2017; Lu et al., 2021; Chen et al., 2021).

The performance for designing a liveness-enforcing PN supervisor is evaluated by three indicators, namely behavioral permissiveness, structural complexity, and computational complexity. In a controlled system, behavioral permissiveness is always evaluated by the quantity of legal markings that are reachable. If a controlled system can reach all the legal markings, its supervisor is said to be maximally permissive, or optimal. A low structural complexity not only represents a simple structure of the supervisor, but also provides its low implementation overheads in terms of both software and hardware costs. A low computational complexity implies that a policy can be efficiently obtained in the sense of computational time. Thus, a significant issue is how to design a supervisor with maximally permissive behavior, a simple structure, and highly computational efficiency, which has been studied by a group of researchers.

Structure analysis (Li and Zhou, 2004; Hu et al., 2015; Wang et al., 2011) and reachability graph analysis (Ghaffari et al., 2003; Huang et al., 2010, 2011; **?**) are two main techniques for analyzing deadlocks based on PNs. Structure analysis is usually efficient by focusing on some special PN structures (siphons and resource circuits, for example). However, a supervisor obtained by doing so is usually suboptimal since some legal markings are prohibited (Li et al., 2008; Li and Zhao, 2008; Li et al., 2011; Tricas et al., 2005). Although the reachability-graph-based methods can lead to an optimal or near-optimal supervisor for a generalized PN (Chen et al., 2011, 2012, 2014; Uzam et al., 2013), they face the disadvantage of enumerating that all reachable markings of a PN model, resulting in the state explosion problem for large-scale systems.

In (Uzam and Zhou, 2006, 2007), a reachability graph are separated into a deadlock-zone (DZ) and a live-zone (LZ), covering all the illegal and legal markings, respectively. An optimal supervisor should ensure the reachability of all markings in the LZ and the non-reachability of markings in the DZ. A first-met bad marking (FBM) is an illegal marking, representing the very first entry from the LZ to the DZ. In (Uzam and Zhou, 2006, 2007; Chen and Li, 2011), deadlock control policies are developed to prevent a system from going to the DZ by controlling every FBM. Such methods solve an integer linear programming problem (ILPP) for every FBM to obtain a set of optimal monitors. Note that one needs to do so not only for every FBM, but also, for each FBM, one requires to consider all legal markings, leading to a large-scale ILPP. Hence, the supervisor is obtained with complex computations and structures.

To solve the above-discussed problem, a VC approach is provided in (Chen et al., 2011). This method can reduce the number of considered markings in finding a controller. By analyzing the relationship among the markings, the markings of activity (operation) places are considered only, aiming to find an MCdS of FBMs and an MCgS of legal markings. It ensures that all FBMs can be forbidden if no FBM in the MCdS of FBMs is reachable. Meanwhile, if the system is able to reach every marking in the MCgS of legal markings, no legal marking is prevented. Thus, one needs to concern only the MCdS of FBMs and the MCgS of legal markings. In this way, one can significantly reduce the computational and structural complexity. However, this method requires to generate all the reachable markings and thus faces the state explosion problem, resulting in the fact that it is inapplicable to large-scale systems.

Similarly, Uzam *et al.* put forward a think-globally-act-locally (TGAL) concept in (Uzam et al., 2016b) to mitigate the state explosion problem for reachability graph analysis. This method adds a temporarily global idle place (GP) into a PN without changing the desired behavior of the original model. It prevents deadlocks iteratively. At the first iteration, it puts one token into the GP and the reachability graph is generated. A controller is designed by adding control places to prevent all deadlocks. Then, increasing the tokens in the GP and the above procedure is repeated to find a controller such that the GP related net is live. This process is repeated until no reachable marking is increased if increasing the tokens in the GP. Finally, the GP is deleted and a live net model is found by adding all the obtained control places.

In (Uzam et al., 2016a), Uzam *et al.* develop an improved TGAL approach, a think-globally-act-locally method with weighted arcs (TGALW). To find a supervisor with weighted arcs, it transforms the original model into a strictly conservative form, denoted as TPNM. In a TPNM, if any transition fires, the number of tokens is kept to be a constant. It claims that a TPNM has the same desired properties as the original PN. The most important gain is that the obtained supervisor by the TPNM is also valid for the original PN model. Finally, this method can reach more legal states than TGAL approach. However, at some iteration steps, it may obtain a suboptimal supervisor due to the fact that some legal markings are forbidden.

With a VC technique applied, our previous work (Li et al., 2017) develops a method to improve the behavioral permissiveness of the supervisors designed by TGAL and TGALW. At every iteration, the considered FBMs and legal markings are reduced by applying the VC technique. By solving a set of ILPPs, a set of control places are computed to prevent all FBMs at each iteration. This method ensures that an optimal controller can be found at each iteration. Thus, the final controlled net has more reachable markings than TGAL and TGALW. However, some PN models are still not optimally controlled by using this method. This is caused by the phenomenon that there are legal markings not appearing at the current iteration, but they may be prohibited by the supervisor obtained in the previous iteration.

This research reports a methodology to the design of maximally permissive supervisors by applying the TGAL method and the VC technique. At each iteration, the legal markings and FBMs are reduced to be two smaller sets by using the VC approach. For each FBM, it guarantees that every place holds the maximal number of tokens by modifying the marking in the MCgS of FBM-related legal markings. By using the modified set of FBM-related legal markings, a number of optimal control places are obtained since no legal marking is prohibited. In this way, we finally obtain an optimally controlled PN model.

To save space, the basics of PNs in (Murata, 1989), the design of a control place by a P-invariant (PI) in (Yamalidou et al., 1996), and the VC approach in (Chen and Li, 2011; Chen et al., 2011) are outlined in (Chen and Li, 2017). The rest of this paper is arranged as follows. Section II develops a number of new definitions necessary and an algorithm for finding the possible legal markings of a system in an iteration process. Section III proposes a method by using the VC approach and TGAL to design a supervisor with maximally permissive behavior. Experimental studies are presented in Section IV to validate the proposed approach. Section V draws conclusions.

2. SYNTHESIS OF M-RELATED SUPER SUCCEED MARKINGS

This section develops two definitions and an algorithm for finding the possible legal markings of a system in an iteration process. This research considers a PN with the following assumption.

Assumption: Let P_A be the set of activity places in a PN model. Then, given a resource place p_i, there exists a minimal PI I_{p_i} such that for all $p \in ||I_{p_i}|| \backslash \{p_i\}$, $p \in P_A$.

This assumption is true for all PNs modeling FMSs in the literature for deadlock prevention.

Definition 1. Let p_i be a resource place and I_{p_i} be a p_i-related PI of (N, M_0) with $P = P^0 \cup P_A \cup P_R$. A P-vector $\mathcal{W}_{p_i}$ is called a p_i-related weight vector, if

$$\mathcal{W}_{p_i}(p) = \begin{cases} I_{p_i}(p) & p \in P_A \\ 0 & otherwise \end{cases} \tag{1}$$

Example *1.* For each resource place p_i, we can always find a PI I_{p_i}. For example, p_{10} is a resource place in Fig. 1. We have a PI related to it, namely $I_{p_{10}} = (0\ 2\ 3\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0)^T$. Then, the p_{10}-related weight vector is $\mathcal{W}_{p_{10}} = (0\ 2\ 3\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0)^T$.

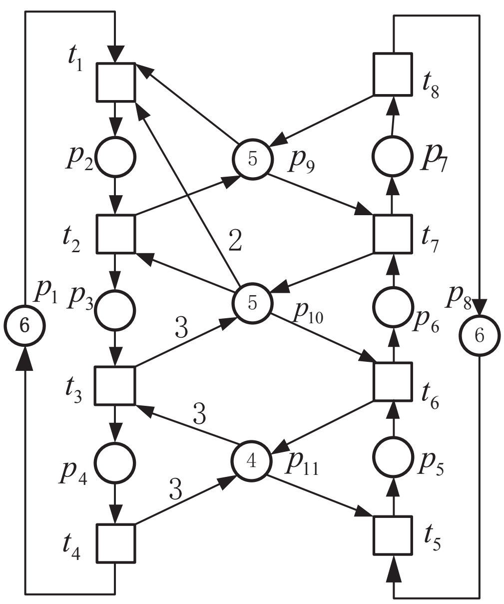

Fig. 1. A PN model.

Definition 2. Let $\mathcal{M}^{\star}_{\text{FBM}}$ represent the MCdS of FBMs, $M \in \mathcal{M}^{\star}_{\text{FBM}}$, and $\mathcal{L}_{\text{M}}$ be the MCgS of M-related legal markings, $M_1 \in \mathcal{L}_{\text{M}}$. A marking M' is called an M-related super succeed marking of M_1 if the three conditions holds:

1) $M' \geq_A M_1$ & $M' \neq M_1$;

2) $(\nexists M'' \in \mathcal{M}^{\star}_{\text{FBM}}) M' \geq_A M''$; and

3) $(\forall p_i \in P_R) \mathcal{W}^T_{p_i} \cdot M' \leq I^T_{p_i} M_0$.

where $I^T_{p_i} M_0$ represents the number of tokens in p_i and p_i-related activity places at the initial marking M_0. A set of M-related super succeed markings of M_1 is denoted as $\mathcal{S}_{M_1}$.

Example *2.* In Fig. 1, marking $M = p_2 + p_3 + 2p_5$ is the only FBM and $M_1 = p_2 + 4p_5$ ($M_1 \in \mathcal{L}_{\text{M}}$) is a legal marking. According to Definition 2, an M-related super succeed markings of M_1 should cover M_1. M contains three activity places: p_2, p_3, and p_5. Thus, we can increase the tokens in p_2, p_3, and p_5 to find the M-related super succeed markings of M_1. It is obvious that p_5 has no more than four tokens due to $M_0(p_{11}) = 4$. By increasing the tokens in p_2 and p_3, two markings $M' = 2p_2 + 4p_5$ and $M'' = p_2 + p_3 + 4p_5$ are obtained, respectively. We have $M' \geq_A M_1$ and $M'' \geq_A M_1$. However, M'' is not an M-related super succeed marking of M_1 due to $M'' \geq_A M$. According to Definition 2, $M' = 2p_2 + 4p_5$ is the only M-related super succeed marking of M_1, since it cannot cover M and satisfies $\mathcal{W}^T_{p_i} \cdot M' \leq I^T_{p_i} M_0$, for all $p_i \in P_R$.

Let $\mathcal{FBM}$ be a set consisting of all markings in $\mathcal{M}^{\star}_{\text{FBM}}$ generated in the iteration process. Next, we propose an algorithm to modify $\mathcal{L}_{\text{M}}$ produced in an iteration process, which can find all M-related super succeed markings.

Algorithm 1 Modification of $\mathcal{L}_{\text{M}}$ produced in an iteration process.

Input: PN (N, M_0) for an FMS with $N = (P^0 \cup P_A \cup P_R, T, F, W)$, an FBM M ($M \in \mathcal{M}^{\star}_{\text{FBM}}$), and $\mathcal{FBM}$.
Output: The modified set $\mathcal{L}'_{\text{M}}$.

foreach $\{M' \in \mathcal{L}_{\text{M}}\}$ **do**
  if $\{\mathcal{W}^T_{p_i} \cdot M' < I^T_{p_i} M_0\}$ **do** /* p_i is a resource place of the PN. */
    foreach $\{p_j\}$ **do** /* p_j is a p_i-related activity place and $M(p_j) \neq 0$. */
      $M_1 = M'$;
      $h = I^T_{p_i} M_0$;
      while $\{h - -\}$
        $M_1(p_j) = M_1(p_j) + 1$;
        if $\{\mathcal{W}^T_{p_i} \cdot M_1 \leq I^T_{p_i} M_0\}$ **do**
          Add the new marking M_1 to the set of $\mathcal{S}_{M'}$. /* $\mathcal{S}_{M'}$ is a set used to store M-related super succeed markings of M' that are found in an iteration process. */
        else
          Break;
      endwhile
    endforeach
  endif
  foreach $\{M_2 \in \mathcal{S}_{M'}\}$ **do**
    if $\{\mathcal{W}^T_{p_k} \cdot M_2 > I^T_{p_k} M_0\}$ **do** /* $p_k \in P_R$. */
      Delete M_2 from $\mathcal{S}_{M'}$.
    elseif $\{M_2 \geq_A M''\}$ **do** /* $M'' \in \mathcal{FBM}$. */

19: Delete M_2 from $\mathcal{S}_{M'}$.
20: **endforeach**
21: **endforeach**
22: Add the markings in $\mathcal{S}_{M'}$ to $\mathcal{L}_{\mathrm{M}}$.
23: Apply the VC approach for $\mathcal{L}_{\mathrm{M}}$ and obtain $\mathcal{L}'_{\mathrm{M}}$.
24: Output $\mathcal{L}'_{\mathrm{M}}$.
25: End.

The above algorithm can obtain a modified set $\mathcal{L}'_{\mathrm{M}}$. For each marking M' in $\mathcal{L}_{\mathrm{M}}$, we first determine whether the tokens in the related resource place p_i ($p_i \in P_R$) are consumed or not. If not consumed, we increase one token in the p_i-related activity place p_j ($p_j \in P^0$ and $M(p_j) \neq 0$) and a marking M_1 can be obtained. It ensures that M_1 can cover M'. This process ends when all tokens in the related resource place p_i are exhaustively consumed. According to Definition 2, M_1 is called an M-related super succeed marking of M' if it cannot cover any marking in $\mathcal{FBM}$ and the total number of tokens owned by p_k-related ($p_k \in P_R$) activity places does not greater than the original number of tokens in p_k. Then, we add M_1 to set $\mathcal{L}_{\mathrm{M}}$. By applying again the VC approach, a new set $\mathcal{L}'_{\mathrm{M}}$ is obtained. Finally, a supervisor can be designed by using the modified $\mathcal{L}'_{\mathrm{M}}$, which ensures that all legal markings are reachable.

By using the modified set $\mathcal{L}'_{\mathrm{M}}$, Equation (10) in (Chen et al., 2011) is transformed as follows:

$$\sum_{i=1}^{n} l_i \cdot (M'(p_i) - M(p_i)) \leq -1, \quad \forall M' \in \mathcal{L}'_{\mathrm{M}} \tag{2}$$

Theorem 1. Let $M \in \mathcal{M}^{\star}_{\mathrm{FBM}}$ be an FBM, l_i be a feasible solution of Equation (2), and β be a non-negative integer. Then, $\sum_{i=0}^{n} l_i \mu_i \leq \beta$ is a maximally permissive linear constraint of M if such a maximally permissive linear constraint exists.

Proof. Suppose that there is a legal marking M' that is prevented by $\sum_{i=0}^{n} l_i \mu_i \leq \beta$, and there does not exist $M'' \geq_M M'$, for all $M'' \in \mathcal{L}'_{\mathrm{M}}$. Thus, marking M' does not satisfy Equation (2) and $M' \notin \mathcal{L}'_{\mathrm{M}}$. We have $\sum_{i=0}^{n} l_i M'(p_i) \geq \beta + 1$, $\beta = \sum_{i=0}^{n} l_i M(p_i) - 1$, and $\sum_{i=0}^{n} l_i (M'(p_i) - M(p_i)) \geq 0$. There are two cases to be considered:

Case 1: $M' \geq_A M$. By Equation (2), we cannot find a feasible solution of l_i, i.e, there does not exist a maximally permissive linear constraint of M. The hypothesis does not hold.

Case 2: M' does not A-cover M. Since M' is prevented by $\sum_{i=0}^{n} l_i \mu_i \leq \beta$ and $M' \notin \mathcal{L}'_{\mathrm{M}}$, there exists a marking $M'' \in \mathcal{L}'_{\mathrm{M}}$ such that $M' \geq_A M''$. First, we have $M' \geq_A M''$ and $M' \neq M''$. Second, M' cannot cover any marking $M_1 \in \mathcal{M}^{\star}_{\mathrm{FBM}}$, since the maximally permissive linear constraint of M_1 does not exist if $M' \geq_A M_1$. Third, for legal marking M', we have $\mathcal{W}^T_{p_i} \cdot M' \leq I^T_{p_i} M_0$ for all $p_i \in P_R$. According to Definition 2, M' is a M-related super succeed marking of M'', which contradicts $M' \notin \mathcal{L}'_{\mathrm{M}}$. Thus, the hypothesis is not true.

Finally, by considering Cases 1 and 2, the conclusion holds.

3. DEADLOCK PREVENTION POLICY

Our previous work (Li et al., 2017) introduces a VC technique to deal with the behavioral permissiveness issue of TGAL and TGALW approaches. At each iteration, the VC technique is applied to reduce FBMs and legal markings that need to be considered. By solving ILPPs, it computes a set of control places in order to prevent all FBMs but no legal marking is forbidden at the current iteration. Thus, the resulting live net has more reachable legal markings compared with TGAL and TGALW. However, for some PN models, one is not able to design an optimal supervisor by using this policy, since some legal markings do not appear at the current iteration but they are prevented by the supervisor obtained in the iteration.

To tackle the above problem in the previous algorithm, an improved iterative deadlock prevention policy is proposed here. It can design control places to ensure that no legal marking is prohibited so as to make the controlled system maximal permissive or it is optimally controlled. The deadlock prevention policy is developed as shown in Algorithm 2.

Algorithm 2 Design of an optimal supervisor by using TGAL approach and the VC technique.

Input: PN (N, M_0) for an FMS with $N = (P^0 \cup P_A \cup P_R, T, F, W)$.
Output: A controlled net $(N^{\alpha}, M_0^{\alpha})$.
1: Calculate the preset and postset of P^0, i.e., $T_I = {}^{\bullet}P^0$ and $T_O = P^{0\bullet}$.
2: Add a global idle place (GP) into the model such that $\mathrm{GP}^{\bullet} = T_O$, ${}^{\bullet}\mathrm{GP} = T_I$, $M_0(\mathrm{GP}) = B$, and the resulting net is denoted as N_B. /* B is the number of tokens in the GP. */
3: **for** $\{B = 1;\ B \leq K;\ B{+}{+}\}$ **do** /* K represents the initial token sum in all idle places. */
4: Calculate the reachability graph of N_B.
5: **if** $\{N_B$ is live$\}$ **do**
6: $B{+}{+}$;
7: Go to Step 3.
8: **else**
9: Calculate the sets of legal markings $\mathcal{M}_L$ and FBMs $\mathcal{M}_{\mathrm{FBM}}$ for N_B.
10: Compute the MCgS $\mathcal{M}^{\star}_L$ and the MCdS $\mathcal{M}^{\star}_{\mathrm{FBM}}$ for legal markings and FBMs, respectively.
11: Add all markings in $\mathcal{M}^{\star}_{\mathrm{FBM}}$ to the set of $\mathcal{FBM}$.
12: **while** $\{\mathcal{M}^{\star}_{\mathrm{FBM}} \neq \emptyset\}$ **do**
13: For an FBM $M \in \mathcal{M}^{\star}_{\mathrm{FBM}}$, compute the MCgS of M-related legal markings $\mathcal{L}_{\mathrm{M}}$.
14: Modify $\mathcal{L}_{\mathrm{M}}$ by using **Algorithm 1** and obtain the modified $\mathcal{L}'_{\mathrm{M}}$.
15: Formulate an ILPP according to Equation (2).
16: Solve the ILPP and obtain l_i's ($i \in \mathbb{N}_A$) and β as the solution.
17: Compute a PI I and its corresponding control place p_s using the method in (Yamalidou et al., 1996).
18: $\mathcal{M}^{\star}_{\mathrm{FBM}} := \mathcal{M}^{\star}_{\mathrm{FBM}} - F_I$.
19: **endwhile**
20: Do the redundancy test (Uzam et al., 2007) to find the necessary control places p_{s_i} ($i = 1, 2, 3, \dots$)

if the number of calculated control places is greater than one.

endif

endfor

Add the necessary control places into (N, M_0) to obtain a controlled net model as (N^α, M_0^α).

Output the controlled net (N^α, M_0^α).

End.

The presented algorithm can obtain a set of optimal monitors in an iterative way. First, we add a place GP into the model that is deadlocks-prone, denoted as N_B (B gives the number of tokens in the GP; $B = 1$ is the initial value). Then, the reachability graph of N_B is generated to check its liveness. If not live, the sets $\mathcal{M}_L$ and $\mathcal{M}_{\mathrm{FBM}}$ are obtained. By applying the VC approach, we obtain the MCgS $\mathcal{M}_L^\star$ and the MCdS $\mathcal{M}_{\mathrm{FBM}}^\star$ for legal markings and FBMs, respectively. For each $M \in \mathcal{M}_{\mathrm{FBM}}^\star$, the set $\mathcal{L}_{\mathrm{M}}$ is modified by using Algorithm 1. With the modified $\mathcal{L}'_{\mathrm{M}}$, optimal control places are computed. By redundancy test, the redundant control places can be removed. When the net N_B is live, the tokens in the GP is increased by one ($B = B + 1$) and the above process is performed again to make N_{B+1} live. At this iteration, the monitors obtained for N_B are also kept in the net N_{B+1}. Continue processes until all legal markings are reachable. Finally, the GP is removed and an optimally controlled PN model is found.

Now, a PN model with 11 places and eight transitions is used to illustrate Algorithm 2, as shown in Fig. 2. It has 125 reachable states with 115 legal markings and 10 FBMs. For this model, we have $P^0 = \{p_1, p_8\}$ as idle places, $P_A = \{p_2 - p_7\}$ as activity places, and $P_R = \{p_9, p_{10}, p_{11}\}$ as resource places.

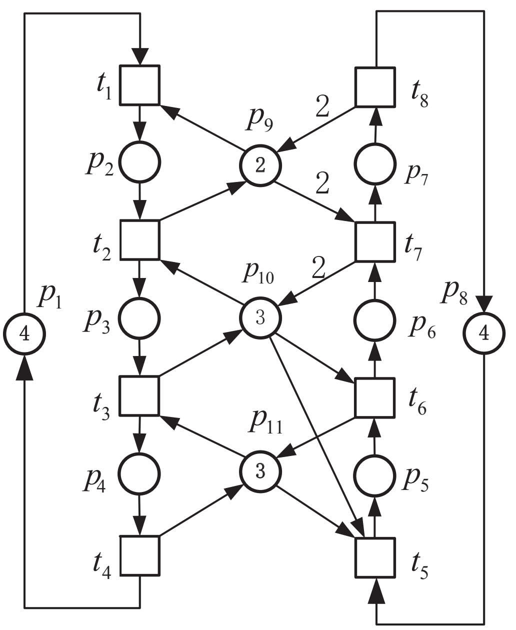

Fig. 2. A PN model.

First, a GP is added into the model. We have ${}^\bullet\mathrm{GP} = T_I = {}^\bullet P^0 = \{t_4, t_8\}$ and $\mathrm{GP}^\bullet = T_O = P^{0\bullet} = \{t_1, t_5\}$. The resulting net is denoted as N_B.

Initially, we put one token into the GP ($B = 1$) to obtain N_1. It can be checked that N_1 is live and has seven legal markings. Then, we increase B by one such that $B = 2$. The net N_2 is also live, including 25 legal markings. When the GP has three tokens ($B = 3$), the related net N_3 is not live. It has two FBMs and 59 legal markings. With the VC technique, we find that $\mathcal{M}_{\mathrm{FBM}}^\star$ and $\mathcal{M}_L^\star$ have two and 32 markings, respectively, where $\mathcal{M}_{\mathrm{FBM}}^\star = \{p_2 + p_5 + p_6, 3p_5\}$. Two FBMs are added into the set $\mathcal{FBM}$.

For $\mathrm{FBM}_1 = p_2 + p_5 + p_6$, the initial MCgS of FBM_1-related legal markings is $\mathcal{L}_{\mathrm{FBM}_1} = \{p_5 + p_6, 2p_2 + p_5, 2p_2 + p_6, p_2 + 2p_5\}$. First, the marking $M_1 = p_5 + p_6$ is selected. For M_1, it holds $\mathcal{W}_{p_9}^T \cdot M_1 < I_{p_9}^T M_0$. The p_9-related activity places are p_2 and p_7. Places p_2, p_5, and p_6 are the FBM_1-related activity places. The activity place p_2 is selected as the only shared place. By increasing the token in it, two markings are obtained, namely $p_2 + p_5 + p_6$ and $2p_2 + p_5 + p_6$. However, none of them is FBM_1-related super succeed marking of M_1 due to $p_2 + p_5 + p_6 \geq_A \mathrm{FBM}_1$ and $2p_2 + p_5 + p_6 \geq_A \mathrm{FBM}_1$, i.e., $\mathcal{S}_{M_1} = \emptyset$.

For the marking $M_2 = 2p_2 + p_5$ in $\mathcal{L}_{\mathrm{FBM}_1}$, we have $\mathcal{W}_{p_{10}}^T \cdot M_2 < I_{p_{10}}^T M_0$. The resource place p_{10}-related activity places are p_3, p_5, and p_6. The FBM_1-related activity places are p_2, p_5, and p_6. Then, the shared activity places p_5 and p_6 are considered. First, we only increase the token in p_5 and generate two markings, namely $2p_2 + 2p_5$ and $2p_2 + 3p_5$. Second, by increasing the token in p_6, a marking $2p_2 + p_5 + p_6$ is obtained. However, we have $2p_2 + 3p_5 \geq_A \mathrm{FBM}_2$ and $2p_2 + p_5 + p_6 \geq_A \mathrm{FBM}_1$. According to Definition 2, $M'_2 = 2p_2 + 2p_5$ is the only FBM_1-related super succeed marking of M_2, since it cannot cover any marking in $\mathcal{FBM}$ and satisfies $\mathcal{W}_{p_i}^T \cdot M'_2 \leq I_{p_i}^T M_0$, for all $p_i \in P_R$. Finally, we have $\mathcal{S}_{M_2} = \{2p_2 + 2p_5\}$.

Similarly, for the marking $M_3 = 2p_2 + p_6$ in $\mathcal{L}_{\mathrm{FBM}_1}$, we have $\mathcal{W}_{p_{10}}^T \cdot M_3 < I_{p_{10}}^T M_0$. Places p_3, p_5, and p_6 are the p_{10}-related activity places. The FBM_1-related activity places are p_2, p_5, and p_6. Obviously, the shared places are p_5 and p_6. First, a marking $2p_2 + p_5 + p_6$ is obtained by only increasing the token in p_5. It is clear that $2p_2 + p_5 + p_6$ is not an FBM_1-related super succeed marking of M_3 thanks to $2p_2 + p_5 + p_6 \geq_A \mathrm{FBM}_1$. Second, we increase the token in p_6 and obtain a marking $M'_3 = 2p_2 + 2p_6$. However, M'_3 is not an FBM_1-related super succeed marking of M_3 since it violates inequation $\mathcal{W}_{p_{10}}^T \cdot M'_3 \leq I_{p_{10}}^T M_0$. There is no FBM_1-related super succeed marking of M_3, i.e., $\mathcal{S}_{M_3} = \emptyset$.

Finally, for the marking $M_4 = p_2 + 2p_5$, we find $\mathcal{W}_{p_9}^T \cdot M_4 < I_{p_9}^T M_0$. The p_9-related activity places are p_2 and p_7. Places p_2, p_5, and p_6 are FBM_1-related activity places. Place p_2 is selected as the only shared activity place. By increasing the token in p_2, a marking $M'_4 = 2p_2 + 2p_5$ is obtained. The marking M'_4 cannot cover any marking in $\mathcal{FBM}$ and also satisfies inequation $\mathcal{W}_{p_i}^T \cdot M'_4 \leq I_{p_i}^T M_0$, for all $p_i \in P_R$. According to Definition 2, M'_4 is an FBM_1-related super succeed marking of M_4, i.e., $\mathcal{S}_{M_4} = \{2p_2 + 2p_5\}$.

Then, we add all markings in $\mathcal{S}_{M_1}$, $\mathcal{S}_{M_2}$, $\mathcal{S}_{M_3}$, and $\mathcal{S}_{M_4}$ into $\mathcal{L}_{\mathrm{FBM}_1}$. By using again the VC technique, $\mathcal{L}_{\mathrm{FBM}_1}$ is modified to $\mathcal{L}'_{\mathrm{FBM}_1} = \{p_5 + p_6, 2p_2 + 2p_5, 2p_2 + p_6\}$. For FBM_1, a PI I_1 is designed to forbid it by meeting Equation (2). I_1 should make the following inequations hold $l_2 \cdot (0 -$

$1)+l_5\cdot(1-1)+l_6\cdot(1-1)\le -1$, $l_2\cdot(2-1)+l_5\cdot(2-1)+l_6\cdot(0-1)\le -1$, and $l_2\cdot(2-1)+l_5\cdot(0-1)+l_6\cdot(1-1)\le -1$. By simplifying the constraints, we have

$$\begin{gathered} -l_2 \le -1 \\ l_2 + l_5 - l_6 \le -1 \text{ and} \\ l_2 - l_5 \le -1 \end{gathered}$$

By solving the above ILPP, a solution is obtained with $l_2 = 1$, $l_5 = 2$, $l_6 = 4$, and $\beta = 6$. A control place p_{s_1} is designed: $\mu_2+2\mu_5+4\mu_6+\mu_{p_{s_1}} = 6$, by using the approach presented in (Yamalidou et al., 1996). By comparison, our previous method in (Li et al., 2017) designs this control place as: $\mu_2+2\mu_5+3\mu_6+\mu_{p_{s_1}} = 5$, which is not an optimal control place since some legal markings are prohibited. Finally, $M_0(p_{s_1}) = 6$, ${}^\bullet p_{s_1} = \{t_2, 4t_7\}$, and $p_{s_1}^\bullet = \{t_1, 2t_5, 2t_6\}$ are obtained. We have $\mathcal{M}^\star_{\mathrm{FBM}} := \mathcal{M}^\star_{\mathrm{FBM}} - \{\mathrm{FBM}_1\}$ and $\mathcal{M}^\star_{\mathrm{FBM}} = \{3p_5\}$.

Next, for $\mathrm{FBM}_2 = 3p_5$, we have $\mathcal{L}_{\mathrm{FBM}_2} = \{2p_5\}$. Thus, the only marking $M = 2p_5$ is selected. For marking M, it holds $\mathcal{W}^T_{p_{10}} \cdot M < I^T_{p_{10}} M_0$. The p_{10}-related activity places are p_3, p_5, p_6 and p_5 is the only activity place with FBM_2-related. The only shared activity place p_5 is considered. However, by increasing one token in p_5, a new generated marking $3p_5$ can cover FBM_2. No marking is the FBM_2-related super succeed marking of M. Finally, we find that $\mathcal{L}_{\mathrm{FBM}_2}$ is not necessarily modified. For the optimal control purpose, a PI I_2 is designed to meet Equation (2). Thus, I_2 has to satisfy $l_5\cdot(2-3)\le -1$. We simplify the constraint as follows:

$$-l_5 \le -1$$

The above ILPP can lead to a solution with $l_5 = 1$ and $\beta = 2$. By using I_2, a control place p_{s_2} is designed, namely $\mu_5 + \mu_{p_{s_2}} = 2$. We have $M_0(p_{s_2}) = 2$, ${}^\bullet p_{s_2} = \{t_6\}$, and $p_{s_2}^\bullet = \{t_5\}$. Finally, $\mathcal{M}^\star_{\mathrm{FBM}} := \mathcal{M}^\star_{\mathrm{FBM}} - \{\mathrm{FBM}_2\}$, and $\mathcal{M}^\star_{\mathrm{FBM}} = \emptyset$. That is, the related net N_3 is live.

Next, we add one more token into the GP ($B = 4$) to analyze the net N_4. Note that the control places p_{s_1} and p_{s_2} are also added into the net N_4. In the next iteration process, the nets N_4 and N_5 are all live. When $B = 6$, the net N_6 can reach 115 legal markings. The iteration is terminated since increasing the token in the GP cannot increase the reachable markings. We remove the GP and add the control places p_{s_1} and p_{s_2} to the original PN. Finally, the PN is optimally controlled. The iteration process of the proposed approach for this example is shown in Table 1, where the first column gives the number of tokens in the GP, and the second and third indicate the control places added in N_B and whether N_B is live or not, respectively. The fourth column presents the number of reachable states of N_B, and the fifth and sixth columns are the numbers of states in the DZ and the LZ, respectively. The seventh column indicates the control places designed and the eighth column is the reachable states of N_B by adding the previously obtained control places.

4. EXPERIMENTAL STUDIES

We use examples to demonstrate the reported techniques. C++ programs are developed to modify $\mathcal{L}_{\mathrm{M}}$ and formulate ILPPs that are solved by Lingo on a desktop with Intel CPU Core 2.8GHz and 4GB-memory.

First, a system from (Uzam et al., 2016a) is considered. It can produce three types of products, namely Parts 1, 2 and 3. Fig. 3 shows the system block diagram. In Fig. 3, six types of resources (R_1–R_6) are used and each type of resource can process two products at a time. An arc labeled by a transition (t_1–t_{18}) means that the resource R_i pointed to by the arc is required if the corresponding transition t_j is triggered. The weight on the arc is the number of resources R_i required. Fig. 4 gives the PN model. In this net model, resource places p_{18}–p_{23} are used to represent six types of resources R_1–R_6, respectively. It has 19,300 reachable markings, including 935 illegal and 18,365 legal markings. The results of using Algorithm 2 are shown in Table 2.

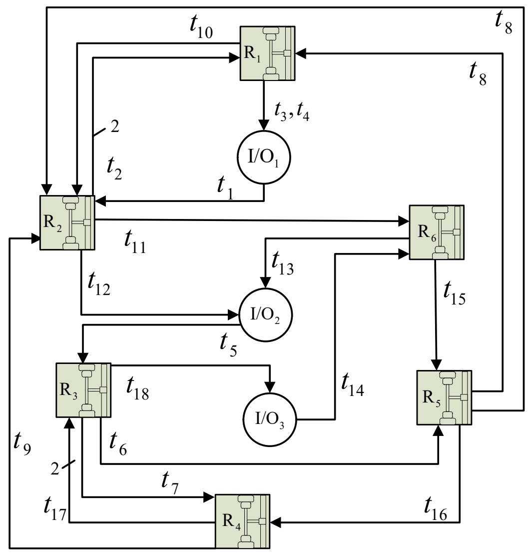

Fig. 3. A system block diagram from (Uzam et al., 2016a).

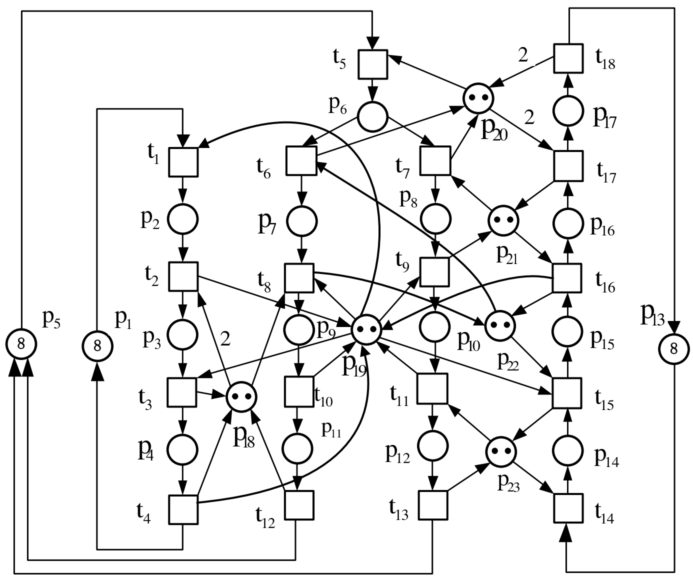

Fig. 4. A PN model from (Uzam et al., 2016a).

When $B = 7$, some FBM_1-related super succeed markings are obtained by using Algorithm 1, where $\mathrm{FBM}_1 = p_2 + p_3+p_6+p_7+p_8+p_{15}+p_{16}$. Similarly, when $B = 8$, we find some FBM_2-related super succeed markings, where $\mathrm{FBM}_2 = p_6 + p_7 + p_8 + p_{10} + 2p_{14} + p_{15} + p_{16}$. We modify $\mathcal{L}_{\mathrm{FBM}_1}$

Table 1. The iteration steps of liveness enforcement for N_B

B	Include p_{s_i}	N_B is live ?	Reachable states	DZ	LZ	p_{s_i}	Reachable states of controlled net
1		Yes	7		7		
2		Yes	25		25		
3		No	59	2	57	p_{s_1} p_{s_2}	57
4	p_{s_1} p_{s_2}	Yes	94		94		
5	p_{s_1} p_{s_2}	Yes	114		114		
6	p_{s_1} p_{s_2}	Yes	115		115		

Table 2. The iteration steps of liveness enforcement for N_B

B	Include p_{s_i}	N_B is live?	Reachable states	DZ	LZ	p_{s_i}	Reachable states of controlled net
1		Yes	15		15		
2		Yes	113		113		
3		No	535	1	534	p_{s_1}	534
4	p_{s_1}	No	1,770	2	1,768	p_{s_2} p_{s_3}	1,768
5	p_{s_1} ... p_{s_3}	No	4,346	4	4,342	p_{s_4} p_{s_5} p_{s_6}	4,342
6	p_{s_1} ... p_{s_6}	No	8,242	1	8,241	p_{s_7}	8,241
7	p_{s_1} ... p_{s_7}	No	12,527	4	12,523	p_{s_8}	12,523
8	p_{s_1} ... p_{s_8}	No	15,893	4	15,889	p_{s_9}	15,889
9	p_{s_1} ... p_{s_9}	Yes	17,666		17,666		
10	p_{s_1} ... p_{s_9}	Yes	18,253		18,253		
11	p_{s_1} ... p_{s_9}	Yes	18,365		18,365		

and $\mathcal{L}_{\mathrm{FBM}_2}$ to find the optimal control places. Finally, the original markings and their $\mathrm{FBM_i}$-related super succeed markings are shown in Table 3, where the first column represents FBMs, and the second and third columns show the original legal markings and their $\mathrm{FBM_i}$-related super succeed markings, respectively.

From Table 3, we find that some different legal markings have the same $\mathrm{FBM_i}$-related super succeed marking. That is to say, the same $\mathrm{FBM_i}$-related super succeed marking can be obtained by different legal markings.

When $B = 11$, the net N_{11} can reach all 18,365 legal markings. We stop the iterations and the GP is removed. Finally, nine control places are found as shown in Table 4. With these nine control places being added, the net is optimally controlled. Table 5 compares the performance of different deadlock control policies for this example.

Next, the proposed approach is applied to an FMS with three processes. Fig. 5 shows its block diagram, including seven types of resources (R_1–R_7) and 20 events (t_1–t_{20}). The capacity of resources R_1, R_5, and R_6 is one and the quantity of other resources is two. The PN model of the system is visualized in Fig. 6. In the model, resources R_1–R_7 are represented by places p_{20}–p_{26}, respectively. There are 24,539 reachable markings, including 19,571 legal markings and 4,968 illegal markings. By applying Algorithm 2, a GP is added into the net and the control places are designed in an iterative way. The iteration process is shown in Table 6.

When $B = 9$, some $\mathrm{FBM_i}$-related ($i = 1, 2, 3$) super succeed markings are found by using Algorithm 1, where $\mathrm{FBM}_1 = p_2 + p_3 + p_4 + p_5 + 2p_7 + p_{14} + p_{15} + p_{17}$, $\mathrm{FBM}_2 = p_2 + p_3 + p_4 + p_6 + p_7 + p_8 + p_{14} + p_{15} + p_{17}$, and $\mathrm{FBM}_3 = p_2 + p_3 + p_6 + p_7 + p_8 + p_{14} + p_{15} + 2p_{17}$. We modify the set of $\mathcal{L}_{\mathrm{FBM_i}}$ to design the optimal control places. Table 7 shows the original legal markings and their $\mathrm{FBM_i}$-related super succeed markings.

Finally, the proposed approach designs 20 control places. With these 20 control places being added into the net model, the resulting controlled net model can make

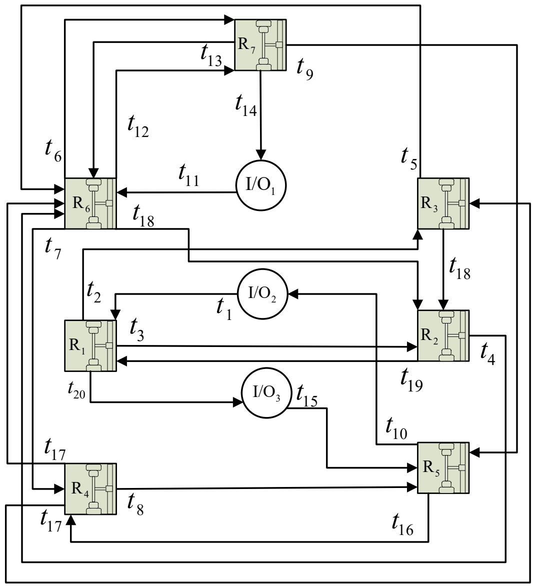

Fig. 5. A system block diagram.

all 19,751 legal markings reachable. Thus, the proposed method can obtain a supervisor with maximally permissive behavior. The obtained control places are shown in Table 8. Table 9 compares the performance of some deadlock prevention policies for this example.

5. CONCLUSION

An improved version of the policy in (Li et al., 2017) is reported to prevent deadlocks, leading to a maximally permissive supervisor for a generalized PN model. A GP is temporarily designed to prevent deadlocks in an iterative way. Compared with the previous approach in (Li et al., 2017), the main difference of this proposed method is that we can find a set of M-related super succeed markings and modify $\mathcal{L}_{\mathrm{M}}$ by using Algorithm 1. It ensures that the control places are designed to prevent all illegal markings while all legal markings are kept. Hence, an optimal supervisor is found. Finally, compared with the study

Table 3. The $\mathrm{FBM_i}$-related super succeed markings

$\mathrm{FBM_i}$	The original markings	The $\mathrm{FBM_i}$-related super succeed markings
$\mathrm{FBM_1}$	$p_2+p_3+p_6+2p_7+2p_{16}$ $p_2+p_3+2p_6+p_7+2p_{16}$ $p_2+2p_6+2p_7+2p_{16}$ $p_3+2p_6+2p_7+2p_{16}$	$p_2+p_3+2p_6+2p_7+2p_{16}$
	$p_2+p_3+p_6+2p_7+p_8+p_{16}$ $p_2+p_3+2p_6+p_7+p_8+p_{16}$ $p_3+2p_6+2p_7+p_8+p_{16}$ $p_2+p_3+2p_6+2p_7+p_8$ $p_2+2p_6+2p_7+p_8+p_{16}$	$p_2+p_3+2p_6+2p_7+p_8+p_{16}$
	$p_2+p_3+p_6+2p_7+2p_8$ $p_2+p_3+2p_6+p_7+2p_8$ $p_3+2p_6+2p_7+2p_8$ $p_2+2p_6+2p_7+2p_8$	$p_2+p_3+2p_6+2p_7+2p_8$
	$2p_2+p_6+2p_7+p_8+p_{16}$ $2p_2+2p_6+p_7+p_8+p_{16}$ $2p_2+2p_6+2p_7+p_{16}$ $p_2+2p_6+2p_7+p_8+p_{16}$ $2p_2+2p_6+2p_7+p_8$	$2p_2+2p_6+2p_7+p_8+p_{16}$
$\mathrm{FBM_2}$	$p_6+2p_7+p_8+2p_{10}+p_{14}+p_{16}$ $2p_6+2p_7+p_8+2p_{10}+p_{16}$ $2p_6+p_7+p_8+2p_{10}+p_{14}+p_{16}$ $2p_6+2p_7+2p_{10}+p_{14}+p_{16}$ $2p_6+2p_7+p_8+2p_{10}+p_{14}$ $2p_6+2p_7+p_8+p_{10}+p_{14}+p_{16}$	$2p_6+2p_7+p_8+2p_{10}+p_{14}+p_{16}$
	$2p_6+2p_7+p_{10}+p_{14}+2p_{16}$	$2p_6+2p_7+2p_{10}+p_{14}+2p_{16}$
	$2p_6+2p_7+2p_8+2p_{10}$ $p_6+2p_7+2p_8+2p_{10}+p_{14}$ $2p_6+p_7+2p_8+2p_{10}+p_{14}$ $2p_6+2p_7+2p_8+p_{10}+p_{14}$	$2p_6+2p_7+2p_8+2p_{10}+p_{14}$
	$p_6+2p_7+p_{10}+2p_{14}+2p_{16}$ $2p_6+p_7+p_{10}+2p_{14}+2p_{16}$ $2p_6+2p_7+p_{10}+2p_{14}+p_{16}$ $2p_6+2p_7+2p_{14}+2p_{16}$	$2p_6+2p_7+p_{10}+2p_{14}+2p_{16}$
	$p_6+2p_7+p_8+p_{10}+2p_{14}+p_{16}$ $2p_6+p_7+p_8+p_{10}+2p_{14}+p_{16}$ $2p_6+2p_7+p_8+p_{10}+2p_{14}$ $2p_6+2p_7+p_8+2p_{14}+p_{16}$	$2p_6+2p_7+p_8+p_{10}+2p_{14}+p_{16}$
	$p_6+2p_7+2p_8+p_{10}+2p_{14}$ $2p_6+p_7+2p_8+p_{10}+2p_{14}$ $2p_6+2p_7+2p_8+2p_{14}$	$2p_6+2p_7+2p_8+p_{10}+2p_{14}$

Table 4. Control places computed of the net shown in Fig. 4 by Algorithm 2

i	I_i	${}^{\bullet}p_{s_i}$	$p_{s_i}^{\bullet}$	$M_0(p_{s_i})$
1	$\mu_2+\mu_3\leq 2$	t_3	t_1	2
2	$\mu_8+\mu_{15}\leq 3$	t_9,t_{16}	t_7,t_{15}	3
3	$\mu_{10}+\mu_{14}\leq 3$	t_{11},t_{15}	t_9,t_{14}	3
4	$\mu_2+\mu_3+\mu_{10}+\mu_{14}\leq 4$	t_3,t_{11},t_{15}	t_1,t_9,t_{14}	4
5	$\mu_2+\mu_3+\mu_8+\mu_{15}\leq 4$	t_3,t_9,t_{16}	t_1,t_7,t_{15}	4
6	$\mu_6+2\mu_8+2\mu_{15}+2\mu_{16}\leq 8$	$t_6,2t_9,2t_{17}$	$t_5,t_7,2t_{15}$	8
7	$\mu_8+\mu_{10}+\mu_{14}+\mu_{15}\leq 5$	t_{11},t_{16}	t_7,t_{14}	5
8	$2\mu_2+2\mu_3+\mu_6+\mu_7+2\mu_8+3\mu_{15}+2\mu_{16}\leq 12$	$2t_3,t_8,2t_9,t_{16},2t_{17}$	$2t_1,t_5,t_7,3t_{15}$	12
9	$\mu_6+\mu_7+2\mu_8+2\mu_{10}+2\mu_{14}+3\mu_{15}+2\mu_{16}\leq 14$	$t_8,2t_{11},t_{16},2t_{17}$	$t_5,t_7,2t_{14},t_{15}$	14

Table 5. Performance comparison of different control policies

Parameters	(Barkaoui et al., 1997)	(Hu et al., 2011)	TGAL in (Uzam et al., 2016b)	TGALW in (Uzam et al., 2016a)	Our previous method in Li et al. (2017)	Proposed method
No. monitors	12	4	9	10	9	9
No. states	10,539	10,613	17,101	18,065	18,248	18,365
Permissiveness (%)	57.38	57.79	93.11	98.37	99.36	100

in (Uzam et al., 2016b), (Uzam et al., 2016a), and our previous work in (Li et al., 2017), only this method can reach all legal markings for a generalized PN model.

However, the presented method also has some drawbacks. First, we need to enumerate a part of reachable markings whose number increases exponentially with respect to the scale of a PN model. Second, the designed supervisor does not have a simple structure, since too many control places are obtained by using this method. Our future work will focus on simplifying the structural complexity of the proposed method and improving the efficiency.

REFERENCES

Barkaoui, K., Chaoui, A., and Zouari, B. (1997). Supervisory control of discrete event systems based on structure theory of Petri nets. In *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, volume 4, 3750–3755. IEEE.

Table 6. The iteration steps of liveness enforcement for N_B

B	Include p_{s_i}	N_B is live?	Reachable states	DZ	LZ	p_{s_i}	Reachable states of controlled net
1		Yes	17		17		
2		Yes	132		132		
3		No	636	5	631	p_{s_1} p_{s_2} p_{s_3} p_{s_4} p_{s_5}	631
4	p_{s_1} ...p_{s_5}	No	2,095	2	2,093	p_{s_6} p_{s_7}	2,093
5	p_{s_1} ...p_{s_7}	No	5,129	4	5,125	p_{s_8} p_{s_9} $p_{s_{10}}$	5,125
6	p_{s_1} ...$p_{s_{10}}$	No	9,639	8	9,631	$p_{s_{11}}$ $p_{s_{12}}$	9,631
7	p_{s_1} ... $p_{s_{12}}$	No	14,370	9	14,361	$p_{s_{13}}$ $p_{s_{14}}$ $p_{s_{15}}$ $p_{s_{16}}$	14,361
8	p_{s_1} ... $p_{s_{16}}$	No	17,730	2	17,728	$p_{s_{17}}$	17,728
9	p_{s_1} ... $p_{s_{17}}$	No	19,207	20	19,187	$p_{s_{18}}$ $p_{s_{19}}$ $p_{s_{20}}$	19,187
10	p_{s_1} ... $p_{s_{20}}$	Yes	19,531		19,531		
11	p_{s_1} ... $p_{s_{20}}$	Yes	19,571		19,571		

Table 7. The FBM$_\mathrm{i}$-related super succeed markings

FBM$_\mathrm{i}$	The original markings	The FBM$_\mathrm{i}$-related super succeed markings
FBM$_1$	$p_2+p_3+p_5+2p_7+p_{14}+p_{15}+p_{17}$	$p_2+p_3+p_5+2p_7+p_{14}+p_{15}+2p_{17}$
	$p_2+2p_3+p_5+p_7+p_{14}+p_{15}+p_{17}$	$p_2+2p_3+p_5+p_7+p_{14}+p_{15}+2p_{17}$
FBM$_2$	$p_2+2p_3+p_4+p_6+2p_7+p_8+p_{17}$ $2p_3+p_4+p_6+2p_7+p_8+p_{14}+p_{17}$ $p_2+p_3+p_4+p_6+2p_7+p_8+p_{14}+p_{17}$ $p_2+2p_3+p_6+2p_7+p_8+p_{14}+p_{17}$ $p_2+2p_3+p_4+2p_7+p_8+p_{14}+p_{17}$ $p_2+2p_3+p_4+p_6+2p_7+p_8+p_{14}$ $p_2+2p_3+p_4+p_6+2p_7+p_{14}+p_{17}$ $p_2+2p_3+p_4+p_6+p_7+p_8+p_{14}+p_{17}$	$p_2+2p_3+p_4+p_6+2p_7+p_8+p_{14}+p_{17}$
	$2p_3+p_4+p_6+2p_7+p_{14}+p_{15}+p_{17}$ $p_2+2p_3+p_4+p_6+2p_7+p_{15}+p_{17}$ $p_2+p_3+p_4+p_6+2p_7+p_{14}+p_{15}+p_{17}$ $p_2+2p_3+p_6+2p_7+p_{14}+p_{15}+p_{17}$ $p_2+2p_3+p_4+2p_7+p_{14}+p_{15}+p_{17}$ $p_2+2p_3+p_4+p_6+2p_7+p_{14}+p_{15}$ $p_2+2p_3+p_4+p_6+p_7+p_{14}+p_{15}+p_{17}$	$p_2+2p_3+p_4+p_6+2p_7+p_{14}+p_{15}+p_{17}$
	$p_2+p_3+p_4+p_6+2p_7+p_8+p_{15}+p_{17}$ $p_2+2p_3+p_6+2p_7+p_8+p_{15}+p_{17}$ $p_2+2p_3+p_4+2p_7+p_8+p_{15}+p_{17}$ $p_2+2p_3+p_4+p_6+2p_7+p_8+p_{15}$ $p_2+2p_3+p_4+p_6+p_7+p_8+p_{15}+p_{17}$	$p_2+2p_3+p_4+p_6+2p_7+p_8+p_{15}+p_{17}$
FBM$_3$	$2p_3+p_6+2p_7+p_8+p_{15}+2p_{17}$ $p_2+p_3+p_6+2p_7+p_8+p_{15}+2p_{17}$ $p_2+2p_3+p_6+2p_7+p_{15}+2p_{17}$ $p_2+2p_3+2p_7+p_8+p_{15}+2p_{17}$ $p_2+2p_3+p_6+2p_7+p_8+p_{15}+p_{17}$ $p_2+2p_3+p_6+2p_7+p_8+2p_{17}$ $p_2+2p_3+p_6+p_7+p_8+p_{15}+2p_{17}$	$p_2+2p_3+p_6+2p_7+p_8+p_{15}+2p_{17}$
	$2p_3+p_6+2p_7+p_8+p_{14}+2p_{17}$ $p_2+p_3+p_6+2p_7+p_8+p_{14}+2p_{17}$ $p_2+2p_3+p_6+2p_7+p_{14}+2p_{17}$ $p_2+2p_3+2p_7+p_8+p_{14}+2p_{17}$ $p_2+2p_3+p_6+2p_7+p_8+p_{14}+p_{17}$ $p_2+2p_3+p_6+2p_7+p_8+2p_{17}$ $p_2+2p_3+p_6+p_7+p_8+p_{14}+2p_{17}$	$p_2+2p_3+p_6+2p_7+p_8+p_{14}+2p_{17}$
	$2p_3+p_6+2p_7+p_{14}+p_{15}+2p_{17}$ $p_2+p_3+p_6+2p_7+p_{14}+p_{15}+2p_{17}$ $p_2+2p_3+2p_7+p_{14}+p_{15}+2p_{17}$ $p_2+2p_3+p_6+2p_7+p_{14}+p_{15}+p_{17}$ $p_2+2p_3+p_6+p_7+p_{14}+p_{15}+2p_{17}$	$p_2+2p_3+p_6+2p_7+p_{14}+p_{15}+2p_{17}$

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

Chen, Y., Li, Y., Li, Z., and Wu, N. (2021). On optimal supervisor design for discrete-event systems modeled with Petri nets via constraint simplification. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*. doi: 10.1109/TSMC.2021.3069201.

Chen, Y. and Li, Z. (2011). Design of a maximally permissive liveness-enforcing supervisor with a compressed supervisory structure for flexible manufacturing systems. *Automatica*, 47(5), 1028–1034.

Chen, Y. and Li, Z. (2012). On structural minimality of optimal supervisors for flexible manufacturing systems. *Automatica*, 48(10), 2647–2656.

Chen, Y. and Li, Z. (2017). Basics of Petri nets in the logic control of resource allocation systems, Xidian Univ., Xi'an, China, Technical Note SCATN01. URL `https://sites.google.com/site/zhiwulisys/notes`.

Chen, Y., Li, Z., and Al-Ahmari, A. (2012). Nonpure Petri net supervisors for optimal deadlock control of flexible manufacturing systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(2), 252–265.

Chen, Y., Li, Z., Barkaoui, K., and Giua, A. (2015). On the enforcement of a class of nonlinear constraints on Petri nets. *Automatica*, 55, 116–124.

Chen, Y., Li, Z., Barkaoui, K., and Uzam, M. (2014). New Petri net structure and its application to optimal supervisory control: Interval inhibitor arcs. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(10), 1384–1400.

Table 8. Control places computed of the net shown in Fig. 6 by Algorithm 2

i	I_i	$^\bullet p_{s_i}$	$p_{s_i}^\bullet$	$M_0(p_{s_i})$
1	$\mu_{11}+\mu_{12}\le 2$	t_{13}	t_{11}	2
2	$\mu_8+\mu_{14}\le 2$	t_8, t_{16}	t_7, t_{15}	2
3	$\mu_4+\mu_{16}\le 2$	t_4, t_{18}	t_3, t_{17}	2
4	$\mu_5+\mu_{15}\le 2$	t_7, t_{17}	t_4, t_{16}	2
5	$\mu_6+\mu_{12}\le 2$	t_6, t_{13}	t_5, t_{12}	2
6	$\mu_4+\mu_{15}\le 3$	t_4, t_{17}	t_3, t_{16}	3
7	$\mu_5+\mu_8+\mu_{14}+\mu_{15}\le 3$	t_8, t_{17}	t_4, t_{15}	3
8	$\mu_4+\mu_8+\mu_{14}+\mu_{15}\le 4$	t_4, t_8, t_{17}	t_3, t_7, t_{15}	4
9	$\mu_4+\mu_5+\mu_{14}+\mu_{15}\le 4$	t_7, t_{17}	t_3, t_{15}	4
10	$\mu_2+\mu_3+\mu_4+2\mu_{16}+\mu_{17}\le 5$	t_4, t_5, t_{18}, t_{19}	$t_1, 2t_{17}$	5
11	$\mu_7+\mu_8+\mu_{11}+\mu_{12}+\mu_{14}+\mu_{15}\le 5$	t_8, t_9, t_{13}, t_{17}	t_6, t_7, t_{11}, t_{15}	5
12	$\mu_6+\mu_7+\mu_8+\mu_{12}+\mu_{14}+\mu_{15}\le 5$	t_8, t_9, t_{13}, t_{17}	t_5, t_7, t_{12}, t_{15}	5
13	$\mu_4+\mu_7+\mu_{11}+\mu_{12}+\mu_{14}+\mu_{15}\le 6$	t_4, t_9, t_{13}, t_{17}	t_3, t_6, t_{11}, t_{15}	6
14	$\mu_4+\mu_6+\mu_7+\mu_{12}+\mu_{14}+\mu_{15}\le 6$	t_4, t_9, t_{13}, t_{17}	t_3, t_5, t_{12}, t_{15}	6
15	$\mu_3+\mu_6+\mu_7+2\mu_8+2\mu_{14}+2\mu_{15}\le 9$	$2t_8, t_9, 2t_{17}$	$t_2, 2t_7, 2t_{15}$	9
16	$\mu_3+\mu_5+\mu_7+\mu_{14}+\mu_{15}\le 6$	t_5, t_7, t_9, t_{17}	t_2, t_4, t_6, t_{15}	6
17	$\mu_3+2\mu_4+\mu_6+\mu_7+2\mu_{14}+2\mu_{15}\le 11$	$2t_4, t_9, 2t_{17}$	$t_2, 2t_3, 2t_{15}$	11
18	$\mu_2+\mu_3+\mu_4+\mu_7+\mu_8+\mu_{14}+\mu_{15}+2\mu_{16}\le 9$	$t_4, t_5, t_8, t_9, 2t_{18}$	$t_1, t_6, t_7, t_{15}, t_{17}$	9
19	$\mu_2+\mu_3+2\mu_4+3\mu_5+\mu_7+3\mu_{14}+3\mu_{15}+\mu_{17}\le 15$	$t_5, 3t_7, t_9, 3t_{17}, t_{19}$	$t_1, t_3, t_4, t_6, 3t_{15}, t_{18}$	15
20	$\mu_2+\mu_3+\mu_4+\mu_6+\mu_7+3\mu_8+3\mu_{14}+3\mu_{15}+\mu_{17}\le 14$	$t_4, 3t_8, t_9, 3t_{17}, t_{19}$	$t_1, 3t_7, 3t_{15}, t_{18}$	14

Table 9. Performance comparison of different control policies

Parameters	TGAL in (Uzam et al., 2016b)	TGALW in (Uzam et al., 2016a)	Our previous method in (Li et al., 2017)	Proposed method
No. monitors	20	23	20	20
No. states	18,512	19,334	19,552	19,571
Permissiveness (%)	94.59	98.79	99.90	100

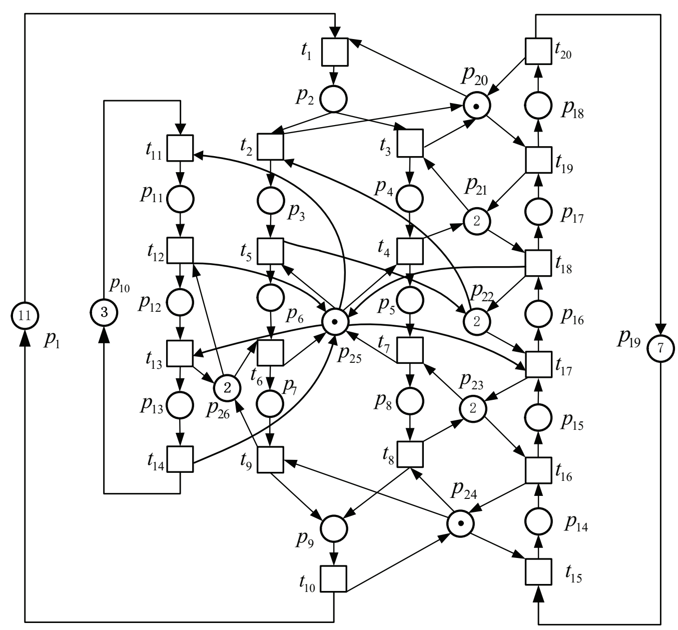

Fig. 6. Net model for an FMS.

Chen, Y., Li, Z., Barkaoui, K., Wu, N., and Zhou, M. (2017). Compact supervisory control of discrete event systems by Petri nets with data inhibitor arcs. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(2), 364–379.

Chen, Y., Li, Z., Khalgui, M., and Mosbahi, O. (2011). Design of a maximally permissive liveness-enforcing Petri net supervisor for flexible manufacturing systems. *IEEE Transactions on automation science and engineering*, 8(2), 374–393.

Coffman, E.G., Elphick, M., and Shoshani, A. (1971). System deadlocks. *ACM Computing Surveys (CSUR)*, 3(2), 67–78.

Cordone, R., Nazeem, A., Piroddi, L., and Reveliotis, S. (2013). Designing optimal deadlock avoidance policies for sequential resource allocation systems through classification theory: existence results and customized algorithms. *IEEE Transactions on Automatic Control*, 58(11), 2772–2787.

Ezpeleta, J., Colom, J.M., and Martinez, J. (1995). A Petri net based deadlock prevention policy for flexible manufacturing systems. *IEEE transactions on robotics and automation*, 11(2), 173–184.

Ghaffari, A., Rezg, N., and Xie, X. (2003). Design of a live and maximally permissive Petri net controller using the theory of regions. *IEEE transactions on robotics and Automation*, 19(1), 137–141.

Hu, H., Liu, Y., and Zhou, M. (2015). Maximally permissive distributed control of large scale automated manufacturing systems modeled with Petri nets. *IEEE Transactions on Control Systems Technology*, 23(5), 2026–2034.

Hu, H. and Zhou, M. (2014). A Petri net-based discrete-event control of automated manufacturing systems with assembly operations. *IEEE Transactions on Control Systems Technology*, 23(2), 513–524.

Hu, H., Zhou, M., and Li, Z. (2011). Supervisor optimization for deadlock resolution in automated manufacturing systems with Petri nets. *IEEE Transactions on Automation Science and Engineering*, 8(4), 794–804.

Huang, Y.S., Jeng, M., Xie, X., and Chung, D.H. (2006). Siphon-based deadlock prevention policy for flexible manufacturing systems. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 36(6), 1248–1256.

Huang, Y.S., Chung, T.H., and Su, P.J. (2010). Synthesis of deadlock prevention policy using Petri nets reachability graph technique. *Asian Journal of Control*, 12(3), 336–346.

Huang, Y.S., Pan, Y.L., and Zhou, M. (2011). Computationally improved optimal deadlock control policy for flexible manufacturing systems. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 42(2), 404–415.

Li, C., Chen, Y., Li, Z., and Barkaoui, K. (2017). Synthesis of liveness-enforcing Petri net supervisors based on a think-globally-act-locally approach and vector covering for flexible manufacturing systems. *IEEE Access*, 5, 16349–16358.

Li, Z., Hu, H., and Wang, A. (2007). Design of liveness-enforcing supervisors for flexible manufacturing systems using Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(4), 517–526.

Li, Z., Wu, N., and Zhou, M. (2011). Deadlock control of automated manufacturing systems based on Petri nets—a literature review. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4), 437–462.

Li, Z. and Zhao, M. (2008). On controllability of dependent siphons for deadlock prevention in generalized Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 38(2), 369–384.

Li, Z. and Zhou, M. (2004). Elementary siphons of Petri nets and their application to deadlock prevention in flexible manufacturing systems. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 34(1), 38–51.

Li, Z. and Zhou, M. (2006a). Clarifications on the definitions of elementary siphons in Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 36(6), 1227–1229.

Li, Z. and Zhou, M. (2006b). Two-stage method for synthesizing liveness-enforcing supervisors for flexible manufacturing systems using Petri nets. *IEEE Transactions on Industrial Informatics*, 2(4), 313–325.

Li, Z., Zhou, M., and Wu, N. (2008). A survey and comparison of Petri net-based deadlock prevention policies for flexible manufacturing systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(2), 173–188.

Lu, Y., Chen, Y., Li, Z., and Wu, N. (2021). An efficient method of deadlock detection and recovery for flexible manufacturing systems by resource flow graphs. *IEEE Transactions on Automation Science and Engineering*. doi:10.1109/TASE.2021.3114498.

Ma, Z., Li, Z., and Giua, A. (2015). Design of optimal Petri net controllers for disjunctive generalized mutual exclusion constraints. *IEEE Transactions on Automatic Control*, 60(7), 1774–1785.

Murata, T. (1989). Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4), 541–580.

Tricas, F., Garcia-Valles, F., Colom, J.M., and Ezpeleta, J. (2005). A Petri net structure-based deadlock prevention solution for sequential resource allocation systems. In *Proceedings of the 2005 IEEE international conference on robotics and automation*, 271–277. IEEE.

Uzam, M. and Zhou, M. (2006). An improved iterative synthesis method for liveness enforcing supervisors of flexible manufacturing systems. *International Journal of Production Research*, 44(10), 1987–2030.

Uzam, M., Gelen, G., and Saleh, T.L. (2016a). Think-globally-act-locally approach with weighted arcs to the synthesis of a liveness-enforcing supervisor for generalized Petri nets modeling FMSs. *Information Sciences*, 363, 235–260.

Uzam, M., Li, Z., and Abubakar, U.S. (2016b). Think globally act locally approach for the synthesis of a liveness-enforcing supervisor of FMSs based on Petri nets. *International Journal of Production Research*, 54(15), 4634–4657.

Uzam, M., Li, Z., and Zhou, M. (2007). Identification and elimination of redundant control places in Petri net based liveness enforcing supervisors of FMS. *The International Journal of Advanced Manufacturing Technology*, 35(1-2), 150–168.

Uzam, M., Zakariyya, R.S., Li, Z., and Gelen, G. (2013). The computation of liveness enforcing supervisors from submodels of a Petri net model of FMSs. In *2013 IEEE International Conference of IEEE Region 10 (TENCON 2013)*, 1–4. IEEE.

Uzam, M. and Zhou, M. (2007). An iterative synthesis approach to Petri net-based deadlock prevention policy for flexible manufacturing systems. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 37(3), 362–371.

Wang, S., Wang, C., and Zhou, M. (2011). Controllability conditions of resultant siphons in a class of Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 42(5), 1206–1215.

Wang, X., Li, Z., and Wonham, W. (2015). Dynamic multiple-period reconfiguration of real-time scheduling based on timed DES supervisory control. *IEEE Transactions on Industrial Informatics*, 12(1), 101–111.

Wu, N. and Zhou, M. (2012). Modeling, analysis and control of dual-arm cluster tools with residency time constraint and activity time variation based on Petri nets. *IEEE Transactions on Automation Science and Engineering*, 9(2), 446–454.

Xing, K., Zhou, M., Wang, F., Liu, H., and Tian, F. (2010). Resource-transition circuits and siphons for deadlock control of automated manufacturing systems. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 41(1), 74–84.

Yamalidou, K., Moody, J., Lemmon, M., and Antsaklis, P. (1996). Feedback control of Petri nets based on place invariants. *Automatica*, 32(1), 15–28.

Zhang, H., Feng, L., Wu, N., and Li, Z. (2017). Integration of learning-based testing and supervisory control for requirements conformance of black-box reactive systems. *IEEE Transactions on Automation Science and Engineering*, 15(1), 2–15.

Zhu, G., Li, Z., and Wu, N. (2018). Model-based fault identification of discrete event systems using partially observed petri nets. *Automatica*, 96, 201–212.

Zhu, G., Li, Z., Wu, N., and Al-Ahmari, A. (2017). Fault identification of discrete event systems modeled by petri nets with unobservable transitions. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 49(2), 333–345.