



ARTICLE

Siphon-Based Divide-and-Conquer Policy for Enforcing Liveness on Petri Net Models of FMS Suffering from Deadlocks or Livelocks

Murat Uzam¹, Bernard Berthomieu², Wei Wei^{3,*}, Yufeng Chen³, Mohammed El-Meligy^{4,5} and Mohamed Abdel Fattah Sharaf⁶

¹Elektrik-Elektronik Mühendisliği Bölümü, Mühendislik-Mimarlık Fakültesi, Yozgat Bozok Üniversitesi, Yozgat, 66100, Türkiye

²Laboratoire d'Analyse et d'Architecture des Systèmes of Centre National de la Recherche Scientifique (LAAS/CNRS) 7, avenue du Colonel Roche, Toulouse, 31077, France

³School of Electro-Mechanical Engineering, Xidian University, Xi'an, 710071, China

⁴Jadara University Research Center, Jadara University, P.O. Box 733, Irbid, 21110, Jordan

⁵Applied Science Research Center, Applied Science Private University, Amman, 11931, Jordan

⁶Department of Industrial Engineering, College of Engineering, King Saud University, Riyadh, 12372, Saudi Arabia

*Corresponding Author: Wei Wei. Email: 19042110618@stu.xidian.edu.cn

Received: 24 June 2025; Accepted: 30 July 2025; Published: 10 November 2025

ABSTRACT: A novel siphon-based divide-and-conquer (SbDaC) policy is presented in this paper for the synthesis of Petri net (PN) based liveness-enforcing supervisors (LES) for flexible manufacturing systems (FMS) prone to deadlocks or livelocks. The proposed method takes an uncontrolled and bounded PN model (UPNM) of the FMS. Firstly, the reduced PNM (RPNM) is obtained from the UPNM by using PN reduction rules to reduce the computation burden. Then, the set of strict minimal siphons (SMSs) of the RPNM is computed. Next, the complementary set of SMSs is computed from the set of SMSs. By the union of these two sets, the superset of SMSs is computed. Finally, the set of subnets of the RPNM is obtained by applying the PN reduction rules to the superset of SMSs. All these subnets suffer from deadlocks. These subnets are then ordered from the smallest one to the largest one based on a criterion. To enforce liveness on these subnets, a set of control places (CPs) is computed starting from the smallest subnet to the largest one. Once all subnets are live, this process provides the LES, consisting of a set of CPs to be used for the UPNM. The live controlled PN model (CPNM) is constructed by merging the LES with the UPNM. The SbDaC policy is applicable to all classes of PNs related to FMS prone to deadlocks or livelocks. Several FMS examples are considered from the literature to highlight the applicability of the SbDaC policy. In particular, three examples are utilized to emphasize the importance, applicability and effectiveness of the SbDaC policy to realistic FMS with very large state spaces.

KEYWORDS: Petri nets; flexible manufacturing systems; deadlock; livelock; liveness-enforcing supervisor

1 Introduction

Nowadays, due to very fast and ever-changing market demands, automated, flexible and agile manufacturing systems are required to answer these demands quickly and efficiently. Deadlocks are an unacceptable system behavior in flexible manufacturing systems (FMS) and occur due to the improper allocation of shared resources such as machines, robots, AGVs, conveyors, etc. In a deadlock state, the whole FMS stops completely, spoiling the use of resources and leading to devastating effects on the operation of these systems [1]. Therefore, in an FMS, the occurrences of deadlocks are not acceptable. A lot of efforts have been put into the study of deadlocks and their resolution in FMS. Petri nets (PN) [2] have been widely utilized for



the study of deadlocks/livelocks, control, and scheduling of FMS [3]. PN based studies to tackle deal with the deadlock problems in FMSs, can be split into three groups as deadlock avoidance [4–8], deadlock detection and recovery [9–11], and deadlock prevention [12–16]. Deadlock prevention policies are preferred over the other methods because the necessary computations are carried out off-line and once. In these methods, to prevent deadlocks/livelocks from occurring, an uncontrolled Petri net model (UPNM) of an FMS prone to deadlocks/livelocks is used to compute a liveness-enforcing supervisor (LES) containing a set of control places (CPs) (also called monitors). CPs contain input arcs, output arcs and initial markings. The controlled live PNM is constructed by merging the LES with the UPNM. A live PN assures operations without deadlocks and livelocks [17–21].

Structural analysis and reachability graph (RG) analysis are two common analysis methods for the study of deadlock problems. Structural analysis is used for the synthesis of LESs by using special PN objects such as siphons or resource-transitions circuits [12,22–25]. The number of siphons grows exponentially w.r.t. the size of a PNM. The RG analysis of a PN in the deadlock prevention studies fully reflects the behavior of an FMS [13,26–29], but requires complete or partial enumeration of the state space [30–34]. In theory, the size of an RG may grow exponentially w.r.t. the size of the PNM. RG-based liveness enforcing methods may provide high permissive behavior, optimal or near-optimal solutions, while in general, structural analysis-based methods provide suboptimal solutions in most cases, because some legal markings cannot be reached. The liveness-enforcing methods relying on special PN objects are applicable to a certain class of PNs.

To evaluate the computed PN-based LESs, there are three criteria: behavioral permissiveness, structural complexity, and computational complexity [35]. The behavioral permissiveness is evaluated by the number of reachable good (legal) markings of the controlled PNM (CPNM). The live CPNM is called to be optimal (maximally permissive) when it is possible to reach all good markings. However, the live CPNM is called suboptimal when some legal markings are not reachable within the live CPNM. The structural complexity is generally evaluated by the number of CPs in an LES. The structural complexity is directly affected by whether or not a CP contains weighted input/output arcs [36]. Therefore, ordinary CPs, whose input/output arc weights are all one, are preferred over the generalized CPs, whose some input/output arc weights are greater than one. Therefore, a low structural complexity provides fewer numbers and a simpler structure of CPs, which means lower implementation expenses in terms of both hardware and software costs. Computational complexity is related to the efficiency of a liveness-enforcing control method. Low computational complexity implies that a control policy can be obtained within a reasonable time, and it can be applied to large-scale PNMs. Therefore, these three criteria have been followed by the research community around the world as three lines of research. In this paper, we are mainly concerned with reducing the computational complexity of computing LESs for large-scale industrial FMSs. In doing so, we also aim to obtain very high permissiveness and low structural complexity.

Divide-and-conquer (DaC) based approaches [37–39] have been proposed to reduce the computational complexity for large-scale systems. In the DaC paradigm, to compute an LES for a UPNM suffering from deadlocks/livelocks, the UPNM is divided into several subnets, which are smaller and easier to handle. When an LES consisting of a set of CPs is computed by using the subnets, a live CPNM is constructed by merging the LES with the UPNM. Therefore, the use of the DaC paradigm in the PN-based liveness enforcing in FMS can be considered as a special direction of research to deal with the computational complexity problem. The work reported in [37] may be considered as the very first DaC method for PN-based liveness enforcing in FMS. In [37], a UPNM is divided into an autonomous subnet, an idle subnet, and several small subnets, i.e., toparchies. An LES, called toparch, is computed for each toparchy. It was shown that the resulting net, called monarch, by composing the toparches derived for the toparchies, can serve as an LES for the given UPNM. It was claimed in [37] that the DaC-based method of [37] computationally outperforms the methods reported

in [12,13,26,40]. The method proposed in [37] deals only with dividing the UPNM into a set of subnets, and the method proposed in [12] is used to compute CPs for the toparchies.

A DaC-based deadlock prevention method was proposed in [39], where decomposition techniques are proposed for deadlock prevention within a class of PNs used to model FMSs. The proposed method first decomposes a PN into two subnets. Next, for each subnet, LESs are computed. After that, two controlled subnets are merged, and another LES is also synthesized from the merged net. Then, the live CPNM is obtained. This policy is based on the RG analysis of the subnets only. There are two drawbacks of this method: it is applicable to S^3PR nets only, and a maximally permissive LES cannot be obtained.

Another DaC method was proposed in [38] for the computation of PN-based LESs for FMSs. A UPNM of an FMS prone to deadlock/livelock is divided into a set of connected subnets by considering all combinations of shared resources. The set of connected subnets contains both live and non-live subnets. Then, subnets with deadlock problems are used to compute an LES to enforce liveness on the UPNM. This technique makes use of RG analysis of a given UPNM and all its subnets, which is carried out by using INA [41]. This PN tool is rather slow when enumerating the whole RG. Besides, INA cannot handle large RGs having a few million states. This fact restricts the application of the method of [38] to realistic FMS with very large RGs. Another drawback is that this method requires considering all live and non-live subnets, which may be a very time-consuming task for large PNMs with a lot of shared resources. The proposed policy in this paper improves the DaC technique of [38] in two ways. The first improvement involves the computation of subnets utilizing a novel and unprecedented approach. To do this, firstly, strict minimal siphons (SMS) of the reduced PNM are computed. Then, from these SMSs, the complementary set of SMSs is computed. With the union of these two sets, supersets of siphons of the reduced PNM are computed. Finally, the set of subnets of the reduced PNM is obtained by applying the PN reduction rules [40] on the superset of siphons. It is important to note that all these subnets suffer from deadlocks. The second improvement is related to the use of the PN tool. In this study, we propose to use TINA [42] as a better alternative, which is much faster than INA. Thanks to the technique reported in [43] for the computation of RGs, live zones (LZs), and deadlock zones (DZs) of a UPNM suffering from deadlocks/livelocks using TINA, the policy proposed in this paper can tackle RGs having more than 100 million states.

A novel siphon-based divide-and-conquer (SbDaC) policy for the computation of LESs for FMSs suffering from deadlocks or livelocks is proposed in this paper. The SbDaC policy is applicable to all classes of PNMs used to model FMSs suffering from deadlock/livelock problems currently accessible in the relevant literature. The SbDaC policy takes a UPNM of an FMS with deadlock/livelock problems as input and provides an LES consisting of a set of CPs to enforce liveness. Thus, the CPNM constructed by merging the LES with the UPNM is live. From a behavioral permissiveness perspective, in general, the CPNM provides optimal or near-optimal behaviour. Since the computed LES contains only ordinary CPs, whose input/output arcs have the weight of 1, the structural complexity of the computed LES is low. Even though it is necessary to compute all SMSs and the RGs of a given UPNM and all its subnets, the SbDaC policy is straightforward, easy to use, and effective. The application of the SbDaC policy for FMS control assures its live operation and high resource utilization and system throughput. Five examples are considered from the literature to show the applicability of the SbDaC policy. In the relevant liveness-enforcing literature, PNMs with very large RGs having more than 100 million states are very rare due to the computational complexity and the demand for very high computational facilities. Since three of the examples used in this paper have very large RGs, this fact highlights the importance, efficiency and applicability of the proposed SbDaC policy.

The paper is organized as follows. Section 2 provides 1. a list of important classes of PNs related to FMS currently available in the literature, 2. some definitions on siphon, complementary set of a siphon and superset of a siphon, 3. PN reduction rules, 4. explanation of RG of a PNM suffering from deadlocks/livelocks.

The proposed SbDaC policy is explained in [Section 3](#). Examples used to show the applicability of the SbDaC policy are provided in the next section. [Section 5](#) presents the conclusions.

2 Basic Concepts

In this paper, the reader is expected to have the basic background information related to PNs [2], and some well-known techniques such as PN reduction, siphon computation, and control synthesis. This section serves as a brief reminder of such concepts used in this paper. In this section, firstly, the classes of PNs related to FMS are highlighted. Then, the concepts related to siphons used in this paper are explained. Next, PN reduction rules are considered. Finally, the RG of an FMS suffering from deadlock problems is studied to unveil the philosophy behind the RG-based deadlock prevention strategy.

2.1 Classes of Petri Nets Related to FMS

The SbDaC policy is applicable to all classes of PNs related to FMS used for the deadlock/livelock studies. Well-known classes of PNs related to FMS include the following nets [44]: PPNs (production Petri nets), S^3PR (systems of simple sequential processes with resources) nets, ES^3PR (extended systems of simple sequential processes with resources) nets, LS^3PR (linear systems of simple sequential processes with resources) nets, ELS^3PR (an extended LS^3PR net), GS^3PR (generalized systems of simple sequential processes with resources) nets, GLS^3PR (generalized linear systems of simple sequential processes with resources) nets, S^3PGR^2 (system of simple sequential processes with general resource requirements) nets, S^3PMR (systems of simple sequential processes with multiple resources) nets, WS^3PR (weighted system of simple sequential processes with resources) nets, WS^3PSR (weighted system of simple sequential processes with several resources) nets, S^4PR (simple systems of simple sequential processes with resources) nets, S^4R (systems of simple sequential processes with shared resources) nets, S^*PR nets, PNR (process nets with resources) nets, RCN-merged nets (resource control nets-merged nets), ERCN-merged nets (extended resource control nets-merged nets), $ERCN^*$ -merged (well-behaved extended resource control nets-merged) nets, G-Systems, Augmented marked graph (AMG) nets, AEMG (augmented extended marked graph) nets, HMG (hierarchical marked graph) nets, HAMG (hierarchical augmented marked graph) nets.

2.2 Siphon, Complementary Set of a Siphon and Superset of a Siphon

P -vector I is called a place invariant (P -invariant) iff $I \neq 0$ and $I^T[N] = 0^T$ hold. P -invariant I is a P -semiflow, if its every entry is non-negative. The set $\|I\| = \{p \in P \mid I[p] \neq 0\}$ is the support of a vector I . A siphon is a non-empty subset of places S , i.e., $S \subseteq P$ is a siphon iff ${}^*S \subseteq S^*$. A trap is a non-empty subset of places S , i.e., $S \subseteq P$ is a trap iff $S^* \subseteq {}^*S$. A siphon (trap) is minimal iff there is no siphon (trap) contained in it as a proper subset [12]. A siphon is said to be a strict minimal siphon (SMS) iff it is minimal and does not contain a marked trap. A siphon refers to a strict minimal siphon (SMS) in this paper unless otherwise stated. If a siphon is the support of a P -semiflow and if it is initially marked, then it can never be emptied. Siphons are very important structural objects that are adopted in deadlock prevention/liveness-enforcing policies for both generalized and ordinary PNMs of FMSs.

For the example S^3PR given in [Fig. 1](#) [45], we have the set of activity places $P_A = \{p2, p3, p4, p6, p7, p8, p9, p10\}$, the set of idle (sink/source) places $P^0 = \{p1, p5\}$, the set of resource places $P_R = \{p11, p12, p13, p14, p15\}$.

Let S be the set of siphons. For the S^3PR given in [Fig. 1](#), there are three siphons $S = \{S_1, S_2, S_3\}$. $S_1 = \{p3, p10, p14, p15\}$, $S_2 = \{p4, p9, p13, p14\}$, $S_3 = \{p4, p10, p13, p14, p15\}$. These siphons are depicted in [Fig. 2](#). ${}^*S_1 = \{t2, t3, t10, t11\}$, $S_1^* = \{t1, t2, t3, t9, t10, t11\}$: ${}^*S_1 \subseteq S_1^*$. ${}^*S_2 = \{t3, t4, t9, t10\}$, $S_2^* = \{t2, t3, t4, t8, t9, t10\}$: ${}^*S_2 \subseteq S_2^*$. ${}^*S_3 = \{t2, t3, t4, t9, t10, t11\}$, $S_3^* = \{t1, t2, t3, t4, t8, t9, t10, t11\}$: ${}^*S_3 \subseteq S_3^*$.

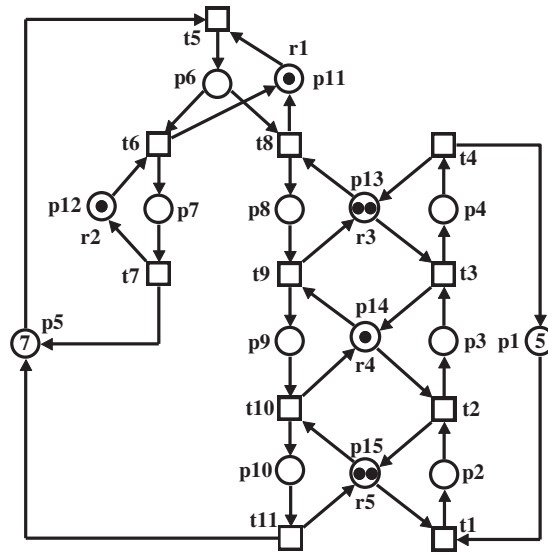
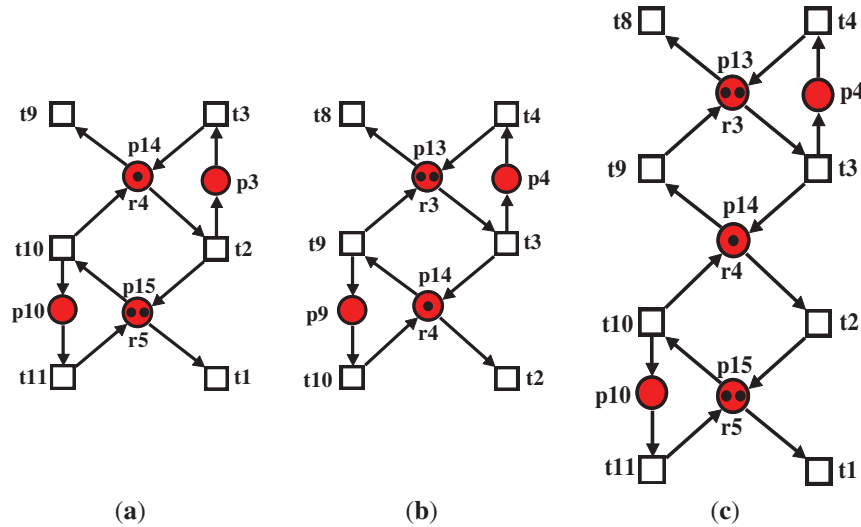


Figure 1: An S³PR net from [45]

Figure 2: Three SMSs $S = \{S_1, S_2, S_3\}$ of the S³PR net shown in Fig. 1. (a) $S_1 = \{p_3, p_{10}, p_{14}, p_{15}\}$; (b) $S_2 = \{p_4, p_9, p_{13}, p_{14}\}$; (c) $S_3 = \{p_4, p_{10}, p_{13}, p_{14}, p_{15}\}$

Let $S \in S$ be an SMS in a Petri net N , where $S = S_A \cup S_R$, $S_R = S \cap P_R = S \setminus S_A$, and $S_A = S \cap P_A = S \setminus S_R$. For example, for $S_1 = \{p_3, p_{10}, p_{14}, p_{15}\}$ we have $S_{1R} = S_1 \cap P_R = S_1 \setminus S_A = \{p_{14}, p_{15}\}$, $S_{1A} = S_1 \cap P_A = S_1 \setminus S_R = \{p_3, p_{10}\}$; for $S_2 = \{p_4, p_9, p_{13}, p_{14}\}$ we have $S_{2R} = S_2 \cap P_R = S_2 \setminus S_A = \{p_{13}, p_{14}\}$, $S_{2A} = S_2 \cap P_A = S_2 \setminus S_R = \{p_4, p_9\}$; for $S_3 = \{p_4, p_{10}, p_{13}, p_{14}, p_{15}\}$ we have $S_{3R} = S_3 \cap P_R = S_3 \setminus S_A = \{p_{13}, p_{14}, p_{15}\}$, $S_{3A} = S_3 \cap P_A = S_3 \setminus S_R = \{p_4, p_{10}\}$.

For $r \in P_R$, $H(r) = {}^{**}r \cap P_A$, the activity (operation) places that use r is called the set of holders of r . For a set of resources $P_R = \{r_1, r_2, \dots, r_m\}$, the set of activity (operation) places whose operations require these resources is denoted by $H(r_1) \cup H(r_2) \cup \dots \cup H(r_m)$ [4]. For example, for the S³PR given in Fig. 1, as we have $P_R = \{p_{11}, p_{12}, p_{13}, p_{14}, p_{15}\}$, the set of holders of $r \in P_R$ are as follows $H(p_{11}) = \{p_6\}$, $H(p_{12}) = \{p_7\}$, $H(p_{13}) = \{p_4, p_8\}$, $H(p_{14}) = \{p_3, p_9\}$, $H(p_{15}) = \{p_2, p_{10}\}$. This means that place p_6 uses p_{11} , place p_7 uses p_{12} , places p_4 and p_8 use p_{13} , places p_3 and p_9 use p_{14} , places p_2 and p_{10} use p_{15} .

$[S] = \cup_{r \in S_R} H(r) \setminus S$ is called the complementary set of siphon S . For example, for the three siphons $S = \{S_1, S_2, S_3\}$: $S_1 = \{p_3, p_{10}, p_{14}, p_{15}\}$, $S_2 = \{p_4, p_9, p_{13}, p_{14}\}$, $S_3 = \{p_4, p_{10}, p_{13}, p_{14}, p_{15}\}$ of the S^3PR shown in Fig. 2, the corresponding complementary sets $[S] = \{[S_1], [S_2], [S_3]\}$ are

$$[S_1] = (H(p_{14}) \cup H(p_{15})) \setminus S_1$$

$$[S_1] = (\{p_3, p_9\} \cup \{p_2, p_{10}\}) \setminus \{p_3, p_{10}, p_{14}, p_{15}\}$$

$$[S_1] = \{p_2, p_3, p_9, p_{10}\} \setminus \{p_3, p_{10}, p_{14}, p_{15}\}$$

$$[S_1] = \{p_2, p_9\}$$

$$[S_2] = (H(p_{13}) \cup H(p_{14})) \setminus S_2$$

$$[S_2] = (\{p_4, p_8\} \cup \{p_3, p_9\}) \setminus \{p_4, p_9, p_{13}, p_{14}\}$$

$$[S_2] = \{p_3, p_4, p_8, p_9\} \setminus \{p_4, p_9, p_{13}, p_{14}\}$$

$$[S_2] = \{p_3, p_8\}$$

$$[S_3] = (H(p_{13}) \cup H(p_{14}) \cup H(p_{15})) \setminus S_3$$

$$[S_3] = (\{p_4, p_8\} \cup \{p_3, p_9\} \cup \{p_2, p_{10}\}) \setminus \{p_4, p_{10}, p_{13}, p_{14}, p_{15}\}$$

$$[S_3] = \{p_2, p_3, p_4, p_8, p_9, p_{10}\} \setminus \{p_4, p_{10}, p_{13}, p_{14}, p_{15}\}$$

$$[S_3] = \{p_2, p_3, p_8, p_9\}$$

The complementary sets of siphons $[S_1]$, $[S_2]$, and $[S_3]$ are depicted in Fig. 3. $\forall S \in S$, $\hat{S} = S \cup [S]$ is defined as the superset of a siphon S . In an ordinary PN, the sum of tokens that exist in a siphon S and the complementary set $[S]$ of S is constant. This means that when a siphon S loses a token, the complementary set $[S]$ of S gains a token and *vice versa* [12]. Thus, $\hat{S} = S \cup [S]$ forms the support of a minimal P -invariant. When all tokens that exist in a siphon S flow into the complementary set $[S]$ of S , S will be empty [12]. For the example S^3PR shown in Fig. 1, the superset of SMSs S , $\hat{S} = \{\hat{S}_1, \hat{S}_2, \hat{S}_3\} = \{S_1 \cup [S_1], S_2 \cup [S_2], S_3 \cup [S_3]\}$ is computed as follows: $\hat{S}_1 = S_1 \cup [S_1] = \{p_3, p_{10}, p_{14}, p_{15}\} \cup \{p_2, p_9\} = \{p_2, p_3, p_9, p_{10}, p_{14}, p_{15}\}$. $\hat{S}_2 = S_2 \cup [S_2] = \{p_4, p_9, p_{13}, p_{14}\} \cup \{p_3, p_8\} = \{p_3, p_4, p_8, p_9, p_{13}, p_{14}\}$. $\hat{S}_3 = S_3 \cup [S_3] = \{p_4, p_{10}, p_{13}, p_{14}, p_{15}\} \cup \{p_2, p_3, p_8, p_9\} = \{p_2, p_3, p_4, p_8, p_9, p_{10}, p_{13}, p_{14}, p_{15}\}$. The supersets of SMSs, $\hat{S}_1$, $\hat{S}_2$, and $\hat{S}_3$ are depicted in Fig. 4. Although the concepts considered in the previous paragraphs on siphons, the complementary set of a siphon, the set of holders of r and the superset of a siphon S are explained using an S^3PR net example, they are also applicable to other classes of PNs related to FMS used for the deadlock/livelock studies currently available in the literature.

2.3 PN Reduction Rules

The LES synthesis approach proposed in this paper makes use of RGs of a given PNM of an FMS suffering from deadlocks/livelocks. To reduce the computation burden, it is important to use the PN reduction approach for PNs with large RGs as explained in [40]. A set of reduction rules include the elimination of self-loop transitions or places, the fusion of parallel transitions or places, fusion of series transitions or places. The PN reduction rules are used to obtain the properties of a complex PNM, while the concerned properties, i.e., boundedness, reversibility and liveness, are preserved. When an LES is computed by using the reduced PNM, it can then be used for the original PNM to enforce liveness. In this paper, PN reduction rules are not only used to obtain the reduced PNM of a given UPNM suffering from deadlock/livelock problems, but also used to obtain reduced subnets of the uncontrolled net model. For example, Fig. 5 depicts the reduced S^3PR net obtained by using the PN reduction rules. The RG of the S^3PR net has 380 markings while its reduced net has only 27 markings.

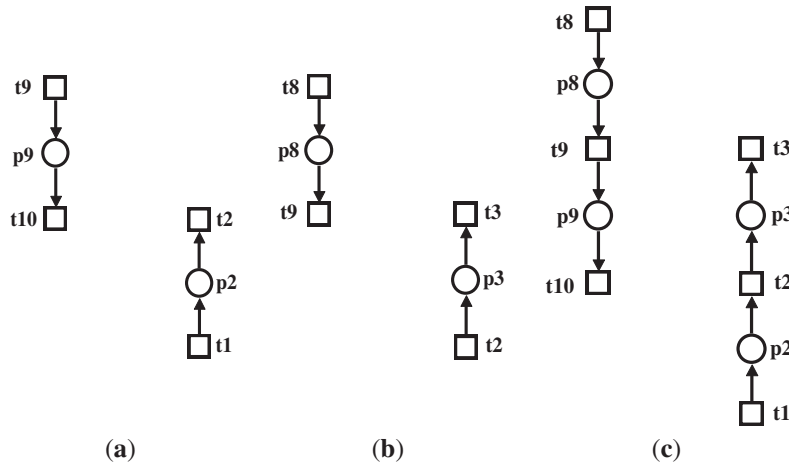


Figure 3: The complementary set of siphons $[S] = \{[S_1], [S_2], [S_3]\}$ of the S^3PR net shown in Fig. 1. (a) $[S_1] = \{p_2, p_9\}$; (b) $[S_2] = \{p_3, p_8\}$; (c) $[S_3] = \{p_2, p_3, p_8, p_9\}$

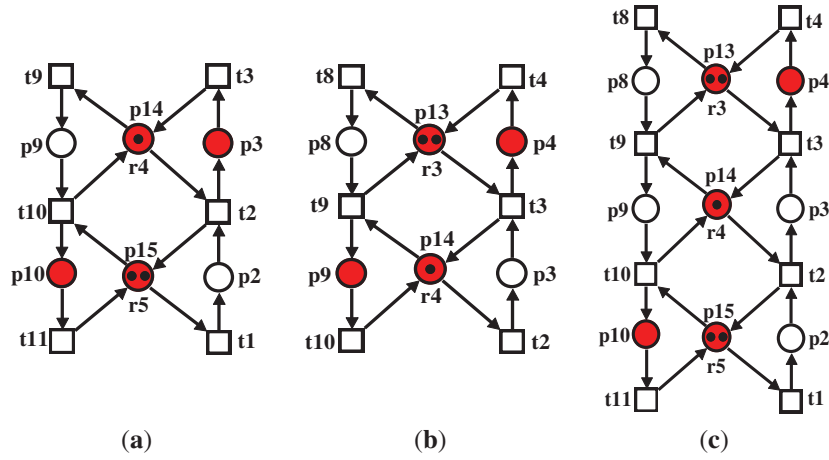


Figure 4: The superset of SMSs $\hat{S} = \{\hat{S}_1, \hat{S}_2, \hat{S}_3\}$ of the S^3PR net shown in Fig. 1. (a) $\hat{S}_1$; (b) $\hat{S}_2$; (c) $\hat{S}_3$

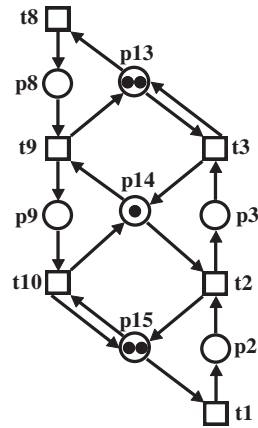


Figure 5: Reduced S^3PR net

2.4 Deadlock/Livelock Studies Based on Reachability Graph (RG) of PNs

In [40], the reachability graph (RG) of an FMS with deadlock problems is divided into a deadlock zone (DZ) and a live zone (LZ). The DZ contains deadlocks, first-met bad markings (FBMs), and bad markings (BMs). An FBM is a marking within the DZ and it represents the first entry from the LZ to the DZ. FBMs and BMs inevitably lead to deadlocks. Deadlocks, FBMs and BMs are all considered to be illegal markings. The LZ consists of all legal markings, from where the initial marking M_0 can be reached. At a deadlock marking no transition can fire. Therefore, it has no successor, which means that this is a dead situation in a system. At a BM, there are firable transitions, i.e., it has successor markings, but the initial marking M_0 cannot be reached from a BM. An FBM is a marking whose ancestor marking is in the LZ, but it is in the DZ. A good marking (GM) can reach M_0 , and its successors can also reach it. A dangerous marking (DM) can reach M_0 , but at least one of its successors cannot reach M_0 . Both good and dangerous markings in the RG must be kept in the controlled system for the sake of optimal control purposes. Therefore, they are considered legal markings. In an RG, when all FBMs are made unreachable by control places (CPs), the system can never go into the DZ from the LZ. In this case, the controlled system runs solely in the LZ, and it is live [46]. Detailed explanations of the deadlock/livelock studies based on RG of PNs are available in [43].

3 A Siphon-Based Divide-and-Conquer (SbDaC) Synthesis Policy for Liveness-Enforcing in FMS

In this section, a novel siphon-based divide-and-conquer (SbDaC) synthesis policy is presented for the computation of LESs consisting of CPs with ordinary arcs for classes of PNs related to FMS suffering from deadlocks or livelocks. Algorithm 1 shows the proposed SbDaC synthesis policy. It is assumed that an uncontrolled bounded PNM (UPNM) of an FMS prone to deadlocks or livelocks is given as input. The PN reduction approach is used to reduce large PNMs to carry out required computations easily, as described in a previous section. The reduced PNM (RPNM) is obtained from the given UPNM (N, M_0) . If the RG of the UPNM is not very large, then the original UPNM can also be used. Given a UPNM of an FMS with deadlock/livelock problems, the aim is to compute an LES consisting of a set of CPs for the UPNM. To compute the LES for an FMS, the RPNM of the system is split into a set of subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{nN}\}$ by using SMSs $S = \{S_1, S_2, \dots, S_n\}$ of the RPNM. SMSs are computed by using INA [41]. To obtain a set of subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{nN}\}$, firstly, the set of SMSs $S = \{S_1, S_2, \dots, S_n\}$ of the RPNM is computed. Secondly, the complementary set of siphons $[S] = \{[S_1], [S_2], \dots, [S_n]\}$ is computed. Next, the superset of SMSs $\hat{S} = \{\hat{S}_1, \hat{S}_2, \dots, \hat{S}_n\} = \{S_1 \cup [S_1], S_2 \cup [S_2], \dots, S_n \cup [S_n]\}$ is computed. These three steps are explained in detail in the previous sections. Finally, the subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{nN}\}$ are obtained from the superset of SMSs $\hat{S}$ by using the PN reduction approach. Then, the RGs of all subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{nN}\}$ are computed, and the numbers of states in the RGs, LZs and DZs of all subnets are defined. These two sets of computations are carried out by using TINA [42] with the guidance of the method explained in [43]. It is important to note that all subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{nN}\}$ suffer from deadlocks. Therefore, it is necessary to consider each subnet starting from the smallest one to the largest one to make it live by computing CPs. Some CPs computed for smaller subnets may be used in the larger subnets.

Algorithm 1: Siphon-based computation of PN-based LES for FMSs

Input: The UPNM (N, M_0) of an FMS suffering from deadlocks or livelocks.

Output: A live CPNM (N_c, M_c) , consisting of (N, M_0) and the set of CPs $\psi = \{C_1, C_2, \dots, C_m\}$.

(1) Define the set of CPs to be computed as ψ .

Initialize $\psi = \{\}$.

(2) Obtain the reduced PNM (RPNM) from (N, M_0) by using the PN reduction rules [40].

(3) Compute the strict minimal siphons (SMS) $S = \{S_1, S_2, \dots, S_n\}$ of the RPNM.

(Continued)

Algorithm 1 (continued)

-
- (4) Compute the complementary set of siphons $[S] = \{[S_1], [S_2], \dots, [S_n]\}$ for $S = \{S_1, S_2, \dots, S_n\}$.
- (5) Compute the superset of SMSs $\hat{S} = \{\hat{S}_1, \hat{S}_2, \dots, \hat{S}_n\} = \{S_1 \cup [S_1], S_2 \cup [S_2], \dots, S_n \cup [S_n]\}$.
- (6) Obtain the subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{nN}\}$ from the superset of SMSs $\hat{S}$ by using the PN reduction rules.
- (7) Compute the RGs of all subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{nN}\}$ and define the number of states in the RGs, LZs and DZs of all subnets.
- (8) Define the set of activity places of each subnet $S_N = \{S_{1N}, S_{2N}, \dots, S_{nN}\}$ and the maximum number of activity places within the subnets.
- (9) for $(i = 1; i \leq I; i = i + 1)/I$ is the maximum number of activity places within the subnets; i is the index value representing the number of activity places in a subnet.*/
- {
- Compute the set of subnets with i number activity places, i.e., S_{Ni} , and the total number of subnets in the set S_{Ni} , i.e., J .
- }
- (10) for $(i = 1; i \leq I; i = i + 1)/I$ is the maximum number of activity places within the subnets; i is the index value representing the number activity places in a subnet.*/
- {
- If $S_{Ni} = \emptyset$, then $i = i + 1$ and goto Step 10.
- Else
- for $(j = 1; j \leq J; j = j + 1)/J$ is the total number of subnets in the set S_{Ni} ; j is the index value showing the current subnet in the set S_{Ni} .*/
- {
- (10.1) If $\psi \neq \emptyset$, and if the set of marked activity places of a PI of any previously computed CP from the set ψ is a subset of $S_{Ni,j}$, i.e., $[PI]_{\text{map}} \subseteq [S_{Ni,j}]_{\text{ap}}$,
- Then Obtain the partially controlled $S_{Ni,j}$, i.e., $pCS_{Ni,j}$, by merging all previously computed CPs, whose $[PI]_{\text{map}} \subseteq [S_{Ni,j}]_{\text{ap}}$, with $S_{Ni,j}$;
- Compute the $RG_{i,j}$ of $pCS_{Ni,j}$,
- If $pCS_{Ni,j}$ is live,
- Then Consider the next $S_{Ni,j}$, i.e., $S_{Ni,(j+1)}$: goto Step 10.1
- Else Define the $LZ_{i,j}$ and the $DZ_{i,j}$ of the $RG_{i,j}$
- Else Use the $RG_{i,j}$ together with its $LZ_{i,j}$ and $DZ_{i,j}$ found in Step 7
- (10.2) Define a PI for each BM within the $DZ_{i,j}$, from the subset of the marked activity places of the BM;
- (10.3) Compute a CP C , for each PI using the simplified invariant-based control technique
- (10.4) Obtain the live controlled subnet $CS_{Ni,j}$ by merging all computed CPs for $S_{Ni,j}$, with $pCS_{Ni,j}$ (respectively, $S_{Ni,j}$)
- (10.5) If the live controlled subnet $CS_{Ni,j}$ includes more than one CP C ,
- Then conduct the redundancy test to define the set of necessary CPs for $CS_{Ni,j}$,
- (10.6) Add computed CPs in the set ψ .
- }
- }
- (11) Carry out the redundancy test [48] for all computed CPs by using the RPNM to find the necessary CPs $\psi = \{C_1, C_2, \dots, C_m\}$.
-

(Continued)

Algorithm 1 (continued)

-
- (12) Obtain the live CPNM (N_c, M_c) by merging the necessary CPs $\psi = \{C_1, C_2, \dots, C_m\}$ with the UPNM (N, M_0) .
- (13) Exit
-

The next step involves the ordering of subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{nN}\}$ from the smallest one to the largest one based on the number of activity places of each subnet. This process provides new sets of subnets as follows: S_{N1} , the set of subnets with 1 activity place, S_{N2} , the set of subnets with 2 activity places, S_{N3} , the set of subnets with 3 activity places, $\dots$, up to the set of subnets with the maximum number of activity places. CPs are computed starting from the smallest nonempty set of subnets to make them live. The RG of each subnet is taken into account with its DZ and LZ. Markings of the DZ (deadlocks, FBMs, BMs) are treated as BMs. They are made unreachable by using the simplified invariant-based control method [40]. In the computations, only the marked activity places of a BM are utilized. To stop the activity places of the BM from being marked, the activity places of the BM are characterized as a place invariant (PI). Therefore, the sum of tokens within the activity places of the PI must be one token fewer than its current value within the BM. A PI is realized by a CP. In this study, the simplified version [40] of the method proposed in [47], is utilized to synthesize a CP from a PI. Computed CPs are included within the set of CPs ψ .

The computation of CPs is carried out with larger subnets. If the set of marked activity places of a PI of any previously computed CP from the set ψ is a subset of the considered subnet $S_{Ni,j}$, i.e., $[PI]_{\text{map}} \subseteq [S_{Ni,j}]_{\text{ap}}$, then the partially controlled subnet $S_{Ni,j}$, called $pCS_{Ni,j}$, is obtained by merging all previously computed CPs whose $[PI]_{\text{map}} \subseteq [S_{Ni,j}]_{\text{ap}}$, with $S_{Ni,j}$. If the partially controlled $S_{Ni,j}$, i.e., $pCS_{Ni,j}$, is live, then the next subnet is considered. If it is not live, then a new set of CPs is also synthesized for the considered subnet $S_{Ni,j}$. This process continues until all subnets are live. Finally, all computed CPs are included within the set ψ . Next, the redundancy test [48] is conducted for all computed CPs by using the RPNM to find the set of necessary CPs $\psi = \{C_1, C_2, \dots, C_m\}$. Finally, the live CPNM (N_c, M_c) is constructed by merging the necessary CPs $\psi = \{C_1, C_2, \dots, C_m\}$ with the PNM (N, M_0) .

The set of necessary CPs $\psi = \{C_1, C_2, \dots, C_m\}$ is structurally simple because CPs consist of only ordinary input/output arcs, i.e., the weight of all arcs within the set of necessary CPs $\psi = \{C_1, C_2, \dots, C_m\}$ is 1. This is because in the synthesis of CPs, each coefficient l_i [47] is set to one. On the contrary, when a CP is merged with the UPNM to forbid a selected BM, this process may lead to loss of some legal markings, as indicated in [27]. This is because in some FMS deadlock/livelock prevention problems, some legal markings could not be reachable when disabling the illegal ones with the added CPs. From the behavioral permissiveness point of view, except for some generalized PNMs with a lot of highly shared resources, the behavioral permissiveness of the obtained controlled models provides an optimal or near-optimal solution.

$[PI]_{\text{map}}$ is called the set of marked activity places of a BM in a DZ. For instance, let us assume, a deadlock is reached when both marked activity places p3 and p8 contain two tokens, i.e., $\mu_3 = 2$ and $\mu_8 = 2$. Then, the PI used to prevent the reachability of this deadlock marking is defined as $\mu_3 + \mu_8 \leq 3$. Therefore, the set of marked activity places of this PI is $[PI]_{\text{map}} = \{p3, p8\}$.

$[S_{Ni,j}]_{\text{ap}}$ is called the set of activity places of a subnet $S_{Ni,j}$. Recall that i is the number of activity places in a subnet and j is the current subnet in the set S_{Ni} . For example, $[S_{N4,1}]_{\text{ap}} = \{p3, p4, p5, p8\}$ means that the first subnet within the set of subnets with four activity places contains the activity places p3, p4, p5 and p8.

The proposed SbDaC policy requires both the complete enumeration of siphons (SMSs) of a given UPNM and the generation of RGs of the given UPNM, the reduced PNM and the subnets. It utilizes well-known techniques such as the siphon computation of the reduced PN, by using INA [41], PN reduction rules [40], the RG computation of the subnets and the reduced PN, by using TINA [42], the computation of the number of states of within the LZ and the DZ, the computation of the markings of all states in the

DZ of the subnets and the reduced PN by using TINA as explained in [43], the redundancy test [48] for the computed CPs, the liveness analysis of the CPNM (N_c , M_c) by using TINA. In addition, BMs of a DZ are defined as PIs, and they are implemented by CPs using the simplified version [40] of the method in [47]. Some of these individual components have been extensively studied and implemented in the popular Petri net tools currently available in the literature. Therefore, further studies may be conducted to reveal the performance comparison between the ones utilized in this paper and the currently available Petri net tools such as Charlie [49,50].

In this study, the RG analysis and computations of both the LZs and the DZs, and the computations of siphons are conducted by using a computer with an Intel(R) Core i5-6200U 2.3 GHz CPU using 12 GB of RAM, running Windows 10 operating system. In addition, the RG analysis and computations of both the LZs and the DZs of the three realistic examples having a very large number of states in their RGs are carried out by another computer with an AMD Ryzen 7 7800X3D 8-Core Processor 4.2 GHz CPU using 128 GB of RAM, running Windows 10 operating system. Conducted experimental studies show that the SbDaC policy is applicable to large-scale FMS-oriented PNMs.

4 Examples

In this section, an illustrative S^3 PMR net and three realistic example nets, namely two S^3 PR nets and an S^3 PGR² net, which belong to different FMS-oriented classes, PNMs are considered from the literature to show the applicability of the SbDaC policy. Especially, the three realistic examples contain very large RGs.

4.1 Illustrative Example— S^3 PMR Net

Fig. 6 depicts an S^3 PMR net of an FMS. Originally, this net model was firstly introduced in [51]. Compared with the original S^3 PMR net, the one considered here, shown in Fig. 6, has two additional arcs, namely Pre(t_9 , p_{26}) and Post(p_{26} , t_8), to make place p_9 bounded. This S^3 PMR net suffers from deadlocks and consists of 26 places, $P = \{p_1-p_{26}\}$ and 20 transitions, $T = \{t_1-t_{20}\}$. In this UPNM, we have $P_R = \{p_{20}-p_{26}\}$, $P_A = \{p_2, p_3, p_4, p_6-p_{13}, p_{15}, p_{16}-p_{19}\}$, and $P^0 = \{p_1, p_5, p_{14}\}$.

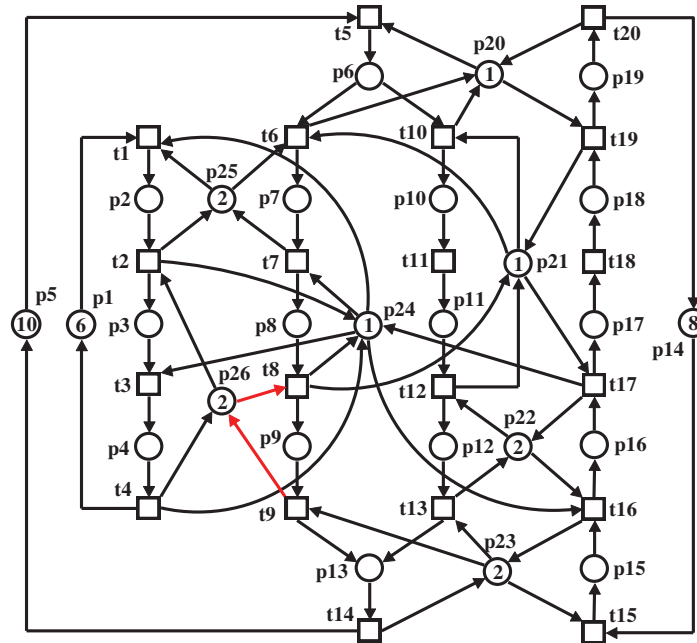


Figure 6: S^3 PMR net of an FMS from [51] with added two arcs prone to deadlocks

Input: The S^3 PMR net (UPNM) (N, M_0) of an FMS suffering from deadlocks depicted in Fig. 6. The RG of the S^3 PMR net has 4691 markings and 17,196 transitions. The LZ and the DZ have 3581 and 1110 states, respectively. Optimally controlled live PNM must reach 3581 good markings.

Step 1. The set of CPs is defined as ψ . $\psi = \{\}$.

Step 2. The reduced S^3 PMR net is depicted in Fig. 7. The RG of the reduced S^3 PMR net contains 1692 markings and 5964 transitions. The LZ and the DZ have 1110 and 582 states, respectively.

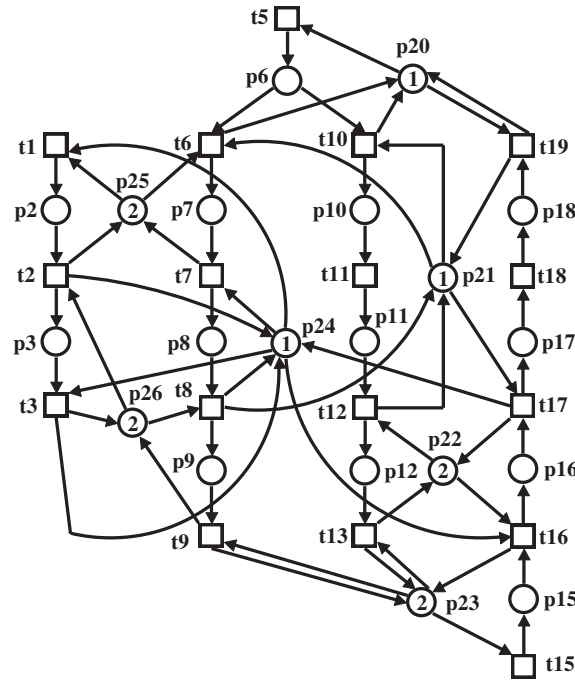


Figure 7: Reduced S^3 PMR net

Note that for brevity in steps 3, 4, and 5, place names are provided without the prefix “p”.

Step 3. By using INA, twenty-four SMSs $S = \{S_1, S_2, \dots, S_{24}\}$ are computed as follows: $S_1 = \{20, 21, 22, 23, 24, 26\}$, $S_2 = \{17, 18, 21, 22, 23, 24, 26\}$, $S_3 = \{10, 11, 20, 21, 23, 24, 26\}$, $S_4 = \{10, 11, 17, 18, 21, 23, 24, 26\}$, $S_5 = \{9, 12, 20, 21, 22, 24, 26\}$, $S_6 = \{9, 12, 17, 18, 21, 22, 24, 26\}$, $S_7 = \{9, 10, 11, 20, 21, 24, 26\}$, $S_8 = \{9, 10, 11, 17, 18, 21, 24, 26\}$, $S_9 = \{16, 23, 24, 26\}$, $S_{10} = \{9, 16, 24, 26\}$, $S_{11} = \{7, 8, 20, 21, 22, 23\}$, $S_{12} = \{7, 8, 12, 20, 21, 22\}$, $S_{13} = \{7, 8, 17, 18, 21, 22, 23\}$, $S_{14} = \{7, 8, 12, 17, 18, 21, 22\}$, $S_{15} = \{7, 8, 10, 11, 20, 21\}$, $S_{16} = \{3, 20, 21, 22, 23, 26\}$, $S_{17} = \{3, 17, 18, 21, 22, 23, 26\}$, $S_{18} = \{16, 22, 23\}$, $S_{19} = \{2, 8, 20, 21, 22, 23, 24\}$, $S_{20} = \{2, 8, 12, 20, 21, 22, 24\}$, $S_{21} = \{2, 8, 17, 18, 21, 22, 23, 24\}$, $S_{22} = \{2, 8, 12, 17, 18, 21, 22, 24\}$, $S_{23} = \{2, 8, 10, 11, 20, 21, 24\}$, $S_{24} = \{2, 8, 10, 11, 17, 18, 21, 24\}$.

Step 4. The twenty-four complementary set of siphons $[S] = \{[S_1], [S_2], \dots, [S_{24}]\}$ are computed as follows: $[S_1] = \{2, 3, 6, 7, 8, 9, 10, 12, 15, 16, 17\}$, $[S_2] = \{2, 3, 7, 8, 9, 10, 12, 15, 16\}$, $[S_3] = \{2, 3, 6, 7, 8, 9, 15, 16, 17\}$, $[S_4] = \{2, 3, 7, 8, 9, 15, 16\}$, $[S_5] = \{2, 3, 6, 7, 8, 10, 16, 17\}$, $[S_6] = \{2, 3, 7, 8, 10, 16\}$, $[S_7] = \{2, 3, 6, 7, 8, 16, 17\}$, $[S_8] = \{2, 3, 7, 8, 16\}$, $[S_9] = \{2, 3, 8, 9, 15\}$, $[S_{10}] = \{2, 3, 8\}$, $[S_{11}] = \{6, 10, 12, 15, 16, 17\}$, $[S_{12}] = \{6, 10, 16, 17\}$, $[S_{13}] = \{10, 12, 15, 16\}$, $[S_{14}] = \{10, 16\}$, $[S_{15}] = \{6, 17\}$, $[S_{16}] = \{6, 7, 8, 9, 10, 12, 15, 16, 17\}$, $[S_{17}] = \{7, 8, 9, 10, 12, 15, 16\}$, $[S_{18}] = \{12, 15\}$, $[S_{19}] = \{6, 7, 10, 12, 15, 16, 17\}$, $[S_{20}] = \{6, 7, 10, 16, 17\}$, $[S_{21}] = \{7, 10, 12, 15, 16\}$, $[S_{22}] = \{7, 10, 16\}$, $[S_{23}] = \{6, 7, 16, 17\}$, $[S_{24}] = \{7, 16\}$.

Step 5. The supersets of SMSs $S, \hat{S} = \{\hat{S}_1, \hat{S}_2, \dots, \hat{S}_{24}\} = \{S_1 \cup [S_1], S_2 \cup [S_2], \dots, S_{24} \cup [S_{24}]\}$ are computed as follows: $\hat{S}_1 = S_1 \cup [S_1] = \{20, 21, 22, 23, 24, 26\} \cup \{2, 3, 6, 7, 8, 9, 10, 12, 15, 16, 17\} = \{2, 3, 6, 7, 8, 9, 10, 12, 15,$

$16, 17, 20, 21, 22, 23, 24, 26\}$. $\widehat{S}_2 = S_2 \cup [S_2] = \{17, 18, 21, 22, 23, 24, 26\} \cup \{2, 3, 7, 8, 9, 10, 12, 15, 16\} = \{2, 3, 7, 8, 9, 10, 12, 15, 16, 17, 18, 21, 22, 23, 24, 26\}$. $\widehat{S}_3 = S_3 \cup [S_3] = \{10, 11, 20, 21, 23, 24, 26\} \cup \{2, 3, 6, 7, 8, 9, 15, 16, 17\} = \{2, 3, 6, 7, 8, 9, 10, 11, 15, 16, 17, 20, 21, 23, 24, 26\}$. $\widehat{S}_4 = S_4 \cup [S_4] = \{10, 11, 17, 18, 21, 23, 24, 26\} \cup \{2, 3, 7, 8, 9, 15, 16\} = \{2, 3, 7, 8, 9, 10, 11, 15, 16, 17, 18, 21, 23, 24, 26\}$. $\widehat{S}_5 = S_5 \cup [S_5] = \{9, 12, 20, 21, 22, 24, 26\} \cup \{2, 3, 6, 7, 8, 10, 16, 17\} = \{2, 3, 6, 7, 8, 9, 10, 12, 16, 17, 20, 21, 22, 24, 26\}$. $\widehat{S}_6 = S_6 \cup [S_6] = \{9, 12, 17, 18, 21, 22, 24, 26\} \cup \{2, 3, 7, 8, 10, 16\} = \{2, 3, 7, 8, 9, 10, 12, 16, 17, 18, 21, 22, 24, 26\}$. $\widehat{S}_7 = S_7 \cup [S_7] = \{9, 10, 11, 20, 21, 24, 26\} \cup \{2, 3, 6, 7, 8, 16, 17\} = \{2, 3, 6, 7, 8, 9, 10, 11, 16, 17, 20, 21, 24, 26\}$. $\widehat{S}_8 = S_8 \cup [S_8] = \{9, 10, 11, 17, 18, 21, 24, 26\} \cup \{2, 3, 7, 8, 16\} = \{2, 3, 7, 8, 9, 10, 11, 16, 17, 18, 21, 24, 26\}$. $\widehat{S}_9 = S_9 \cup [S_9] = \{16, 23, 24, 26\} \cup \{2, 3, 8, 9, 15\} = \{2, 3, 8, 9, 15, 16, 23, 24, 26\}$. $\widehat{S}_{10} = S_{10} \cup [S_{10}] = \{9, 16, 24, 26\} \cup \{2, 3, 8\} = \{2, 3, 8, 9, 16, 24, 26\}$. $\widehat{S}_{11} = S_{11} \cup [S_{11}] = \{7, 8, 20, 21, 22, 23\} \cup \{6, 10, 12, 15, 16, 17\} = \{6, 7, 8, 10, 12, 15, 16, 17, 20, 21, 22, 23\}$. $\widehat{S}_{12} = S_{12} \cup [S_{12}] = \{7, 8, 12, 20, 21, 22\} \cup \{6, 10, 16, 17\} = \{6, 7, 8, 10, 12, 16, 17, 20, 21, 22\}$. $\widehat{S}_{13} = S_{13} \cup [S_{13}] = \{7, 8, 17, 18, 21, 22, 23\} \cup \{10, 12, 15, 16\} = \{7, 8, 10, 12, 15, 16, 17, 18, 21, 22, 23\}$. $\widehat{S}_{14} = S_{14} \cup [S_{14}] = \{7, 8, 12, 17, 18, 21, 22\} \cup \{10, 16\} = \{7, 8, 10, 12, 16, 17, 18, 21, 22\}$. $\widehat{S}_{15} = S_{15} \cup [S_{15}] = \{7, 8, 10, 11, 20, 21\} \cup \{6, 17\} = \{6, 7, 8, 10, 11, 17, 20, 21\}$. $\widehat{S}_{16} = S_{16} \cup [S_{16}] = \{3, 20, 21, 22, 23, 26\} \cup \{6, 7, 8, 9, 10, 12, 15, 16, 17\} = \{3, 6, 7, 8, 9, 10, 12, 15, 16, 17, 20, 21, 22, 23, 26\}$. $\widehat{S}_{17} = S_{17} \cup [S_{17}] = \{3, 17, 18, 21, 22, 23, 26\} \cup \{7, 8, 9, 10, 12, 15, 16\} = \{3, 7, 8, 9, 10, 12, 15, 16, 17, 18, 21, 22, 23, 26\}$. $\widehat{S}_{18} = S_{18} \cup [S_{18}] = \{16, 22, 23\} \cup \{12, 15\} = \{12, 15, 16, 22, 23\}$. $\widehat{S}_{19} = S_{19} \cup [S_{19}] = \{2, 8, 20, 21, 22, 23, 24\} \cup \{6, 7, 10, 12, 15, 16, 17\} = \{2, 6, 7, 8, 10, 12, 15, 16, 17, 20, 21, 22, 23, 24\}$. $\widehat{S}_{20} = S_{20} \cup [S_{20}] = \{2, 8, 12, 20, 21, 22, 24\} \cup \{6, 7, 10, 16, 17\} = \{2, 6, 7, 8, 10, 12, 16, 17, 20, 21, 22, 24\}$. $\widehat{S}_{21} = S_{21} \cup [S_{21}] = \{2, 8, 17, 18, 21, 22, 23, 24\} \cup \{7, 10, 12, 15, 16\} = \{2, 7, 8, 10, 12, 15, 16, 17, 18, 21, 22, 23, 24\}$. $\widehat{S}_{22} = S_{22} \cup [S_{22}] = \{2, 8, 12, 17, 18, 21, 22, 24\} \cup \{7, 10, 16\} = \{2, 7, 8, 10, 12, 16, 17, 18, 21, 22, 24\}$. $\widehat{S}_{23} = S_{23} \cup [S_{23}] = \{2, 8, 10, 11, 20, 21, 24\} \cup \{6, 7, 16, 17\} = \{2, 6, 7, 8, 10, 11, 16, 17, 20, 21, 24\}$. $\widehat{S}_{24} = S_{24} \cup [S_{24}] = \{2, 8, 10, 11, 17, 18, 21, 24\} \cup \{7, 16\} = \{2, 7, 8, 10, 11, 16, 17, 18, 21, 24\}$

Step 6. The subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{24N}\}$ depicted in Fig. 8 are obtained from the superset of SMSs $S, \hat{S} = \{\widehat{S}_1, \widehat{S}_2, \dots, \widehat{S}_{24}\}$ by using the PN reduction rules.

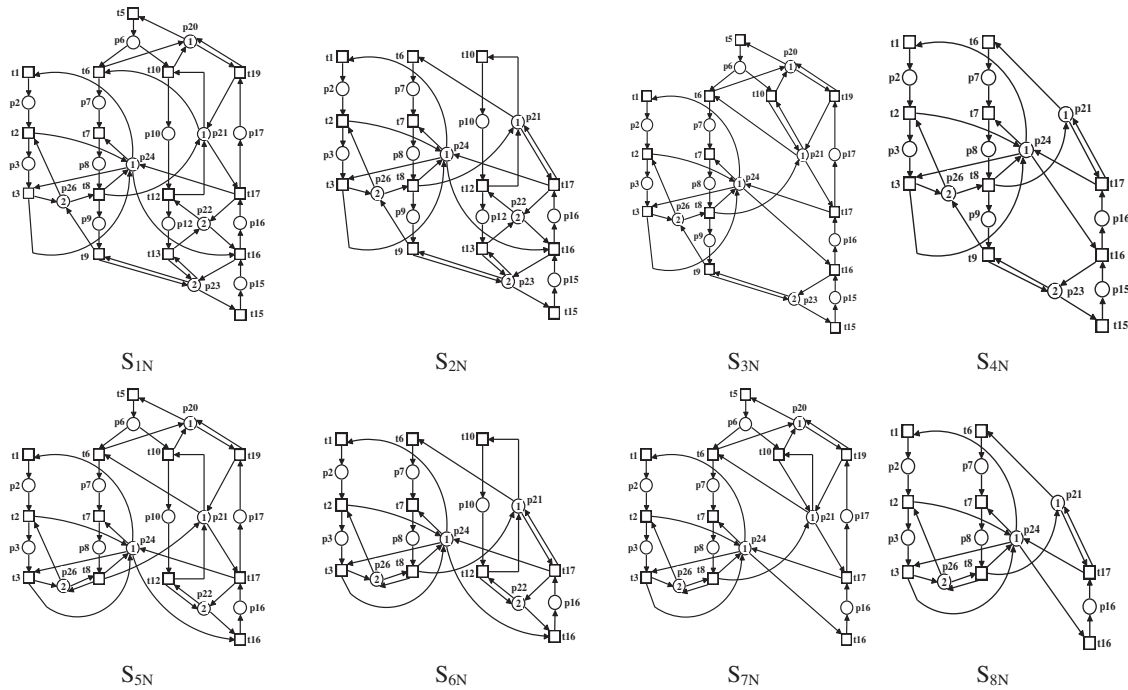


Figure 8: (Continued)

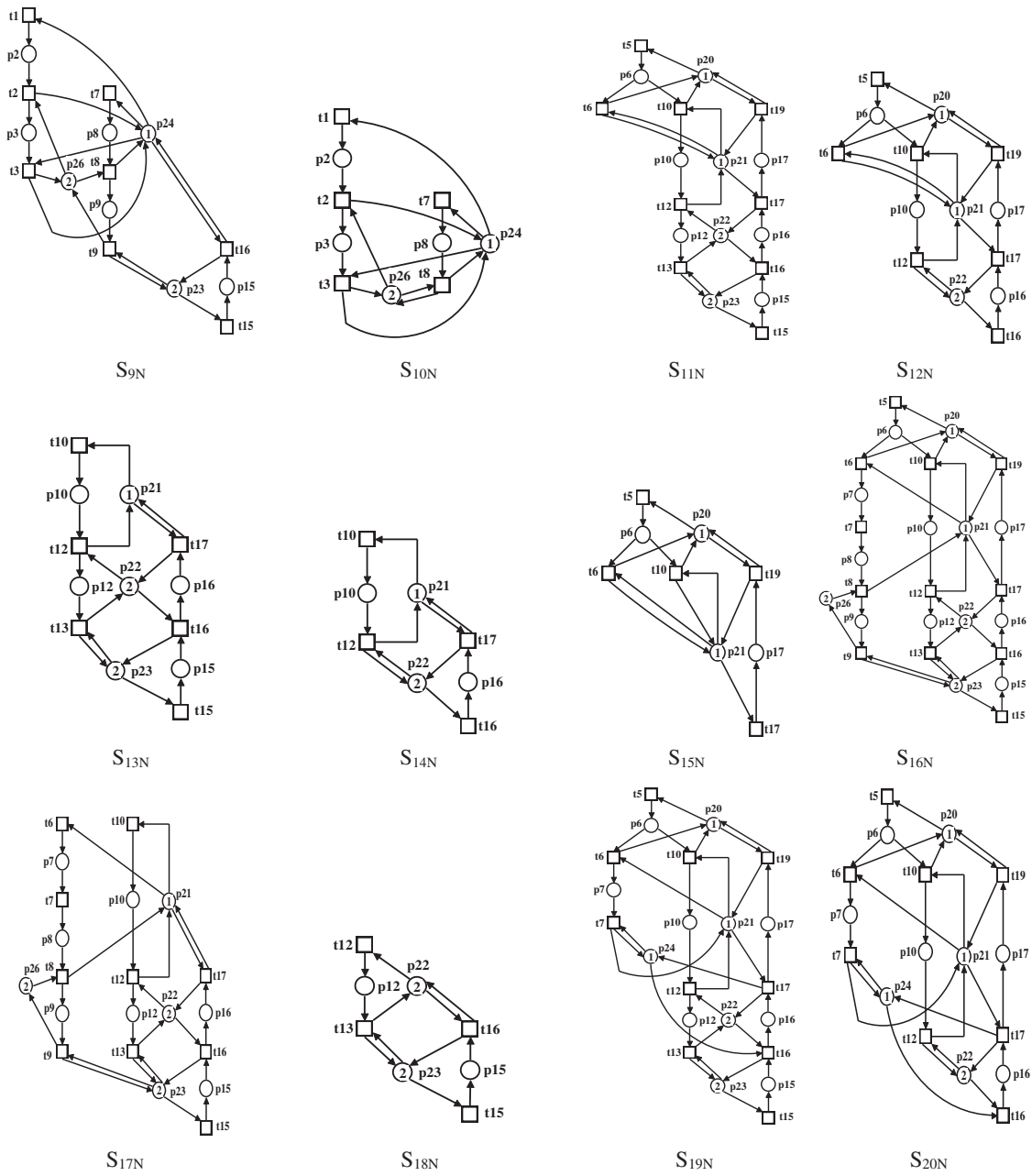


Figure 8: (Continued)

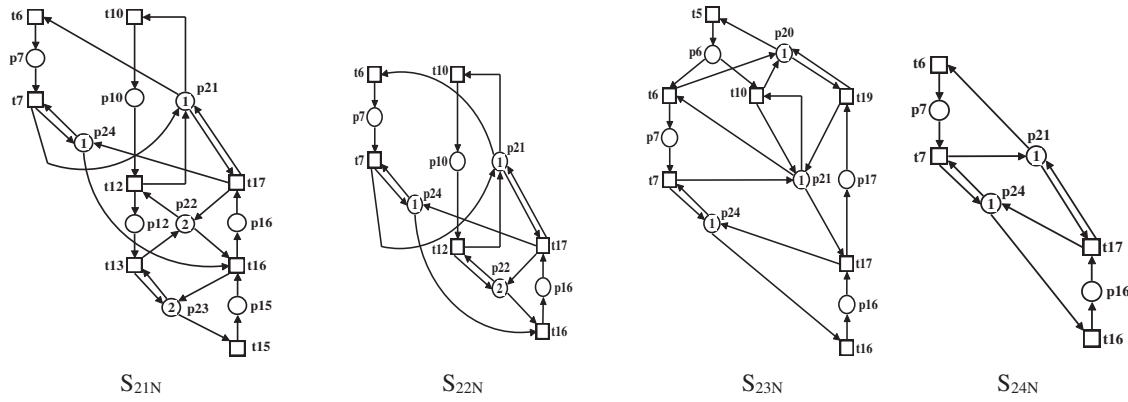


Figure 8: The subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{24N}\}$

Step 7. The RGs of all subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{24N}\}$ are computed, and the numbers of states in the RGs, LZs and DZs of all subnets are defined. The number of markings in the RG, DZ and LZ of the subnets $S_N = \{S_{1N}, S_{2N}, \dots, S_{24N}\}$ are depicted in [Table 1](#).

Table 1: The number of states (# S) in the RG, LZ and DZ of the subnets $S_{1N}, S_{2N}, \dots, S_{24N}$

# S	S_{1N}	S_{2N}	S_{3N}	S_{4N}	S_{5N}	S_{6N}	S_{7N}	S_{8N}	S_{9N}	S_{10N}	S_{11N}	S_{12N}
RG	1188	486	360	126	78	30	60	21	54	9	102	18
LZ	790	352	233	92	54	23	38	15	44	7	75	13
DZ	398	134	127	34	24	7	22	6	10	2	27	5

# S	S_{13N}	S_{14N}	S_{15N}	S_{16N}	S_{17N}	S_{18N}	S_{19N}	S_{20N}	S_{21N}	S_{22N}	S_{23N}	S_{24N}
RG	36	6	4	522	216	9	114	16	45	6	12	4
LZ	30	5	3	414	188	8	80	12	35	5	8	3
DZ	6	1	1	108	28	1	34	4	10	1	4	1

Step 8. The set of activity places of each subnet $S_N = \{S_{1N}, S_{2N}, \dots, S_{24N}\}$ is defined as depicted in [Table 2](#). For instance, the set of activity places of subnets S_{14N} , S_{13N} , and S_{6N} are as follows $[S_{14N}]_{ap} = \{p_{10}, p_{16}\}$, $[S_{13N}]_{ap} = \{p_{10}, p_{12}, p_{15}, p_{16}\}$, $[S_{6N}]_{ap} = \{p_2, p_3, p_7, p_8, p_{10}, p_{16}\}$, with $|[S_{14N}]_{ap}| = 2$, $|[S_{13N}]_{ap}| = 4$, $|[S_{6N}]_{ap}| = 6$. The maximum number of activity places within the subnets is $I = |[S_{1N}]_{ap}| = 11$, since $[S_{1N}]_{ap} = \{p_2, p_3, p_6, p_7, p_8, p_9, p_{10}, p_{12}, p_{15}, p_{16}, p_{17}\}$.

Table 2: Subnets with their activity places (AP) and the total number of APs

S_N	Activity Places (AP) of subnets											# AP
	p2	p3	p6	p7	p8	p9	p10	p12	p15	p16	p17	
S_{14N}							1			1		2
S_{15N}			1								1	2
S_{18N}								1	1			2
S_{24N}				1						1		2
S_{10N}	1	1			1							3

(Continued)

Table 2 (continued)

S_N	Activity Places (AP) of subnets											# AP
	p2	p3	p6	p7	p8	p9	p10	p12	p15	p16	p17	
S_{22N}				1			1			1		3
S_{12N}			1				1			1	1	4
S_{13N}							1	1	1	1		4
S_{23N}			1	1						1	1	4
S_{8N}	1	1		1	1					1		5
S_{9N}	1	1			1	1			1			5
S_{20N}			1	1			1			1	1	5
S_{21N}				1			1	1	1	1		5
S_{6N}	1	1		1	1		1			1		6
S_{11N}			1				1	1	1	1	1	6
S_{4N}	1	1		1	1	1			1	1		7
S_{7N}	1	1	1	1	1					1	1	7
S_{17N}				1	1	1	1	1	1	1		7
S_{19N}			1	1			1	1	1	1	1	7
S_{5N}	1	1	1	1	1		1			1	1	8
S_{2N}	1	1		1	1	1	1	1	1	1		9
S_{3N}	1	1	1	1	1	1			1	1	1	9
S_{16N}			1	1	1	1	1	1	1	1	1	9
S_{1N}	1	1	1	1	1	1	1	1	1	1	1	11

Step 9. for ($i = 1; i \leq (I = 11); i = i++$)

{

The set of subnets (S_N s) with 1 activity place (AP), $S_{N1} = \{\}$, $J = |S_{N1}| = 0$.

$i = 2$

The set of S_N s with 2 APs, $S_{N2} = \{S_{14N}, S_{15N}, S_{18N}, S_{24N}\}$, $J = |S_{N2}| = 4$.

$i = 3$

The set of S_N s with 3 APs, $S_{N3} = \{S_{10N}, S_{22N}\}$, $J = |S_{N3}| = 2$.

$i = 4$

The set of S_N s with 4 APs, $S_{N4} = \{S_{12N}, S_{13N}, S_{23N}\}$, $J = |S_{N4}| = 3$.

$i = 5$

The set of S_N s with 5 APs, $S_{N5} = \{S_{8N}, S_{9N}, S_{20N}, S_{21N}\}$, $J = |S_{N5}| = 4$.

$i = 6$

The set of S_N s with 6 APs, $S_{N6} = \{S_{6N}, S_{11N}\}$, $J = |S_{N6}| = 2$.

$i = 7$

The set of S_N s with 7 APs, $S_{N7} = \{S_{4N}, S_{7N}, S_{17N}, S_{19N}\}$, $J = |S_{N7}| = 4$.

$i = 8$

The set of S_N s with 8 APs, $S_{N8} = \{S_{5N}\}$, $J = |S_{N8}| = 1$.

$i = 9$

The set of S_N s with 9 APs, $S_{N9} = \{S_{2N}, S_{3N}, S_{16N}\}$, $J = |S_{N9}| = 3$.

$i = 10$

The set of S_N s with 10 APs, $S_{N10} = \{\}$, $J = |S_{N10}| = 0$.

$i = 11$

The set of S_N s with 11 APs, $S_{N11} = \{S_{1N}\}$, $J = |S_{N11}| = 1$.

}

Step 10. for ($i = 1$; $i \leq (I = 11)$; $i = i++$)

Table 3 provides a summary of Step 10 applied for the S^3 PMR net. For subnets S_{14N} , S_{15N} , S_{18N} , S_{24N} , and S_{10N} , the RGs $RG_{S_{14N}}$, $RG_{S_{15N}}$, $RG_{S_{18N}}$, $RG_{S_{24N}}$, and $RG_{S_{10N}}$ indicated in Table 1 are used respectively for the synthesis of CPs C1, C2, C3, C4, C5 and C6. For subnet S_{14N} , the $RG_{S_{14N}}$ consists of six states, with five good states in the LZ and BM_1 in the DZ. The BM_1 is obtained as follows: $BM_1 = p_{10} + 2p_{16}$. To stop BM_1 from being reached $PI_1 = \mu_{10} + \mu_{16} \leq 2$ is established with $[PI_1]_{\text{map}} = \{p_{10}, p_{16}\}$. The CP C1 is computed to enforce PI_1 . Likewise, for subnets S_{15N} , S_{18N} , S_{24N} , and S_{10N} , BMs $BM_2 = p_6 + p_{17}$, $BM_3 = 2p_{12} + 2p_{15}$, $BM_4 = p_7 + p_{16}$, $BM_5 = p_2 + 2p_3$, and $BM_6 = 2p_3 + p_8$ are obtained. Place invariants $PI_2 = \mu_6 + \mu_{17} \leq 1$, $PI_3 = \mu_{12} + \mu_{15} \leq 3$, $PI_4 = \mu_7 + \mu_{16} \leq 1$, $PI_5 = \mu_2 + \mu_3 \leq 2$, $PI_6 = \mu_3 + \mu_8 \leq 2$ are established with $[PI_2]_{\text{map}} = \{p_6, p_{17}\}$, $[PI_3]_{\text{map}} = \{p_{12}, p_{15}\}$, $[PI_4]_{\text{map}} = \{p_7, p_{16}\}$, $[PI_5]_{\text{map}} = \{p_2, p_3\}$, $[PI_6]_{\text{map}} = \{p_3, p_8\}$, and finally CPs C2, C3, C4, C5 and C6 are computed. It is verified that the controlled subnet S_{14N} , (respectively, S_{15N} , S_{18N} , S_{24N} , S_{10N}) constructed by merging the CP C1 (C2, C3, C4, C5 and C6) with the uncontrolled S_{14N} (respectively, S_{15N} , S_{18N} , S_{24N} , S_{10N}) is live with 5 (respectively, 3, 8, 3, 7) good states. When $i = 3, j = 2$, we have $\psi = \{C1, C2, C3, C4, C5, C6\}$. Since $[S_{22N}]_{\text{ap}} = \{p_7, p_{10}, p_{16}\}$ we have $[PI_1]_{\text{map}} \subseteq [S_{22N}]_{\text{ap}}$ and $[PI_4]_{\text{map}} \subseteq [S_{22N}]_{\text{ap}}$. Thus, pCS_{22N} is constructed by merging C1 and C4 with S_{22N} . The RG of the pCS_{22N} contains 5 good states in the $LZ_{S_{22N}}$ and no states in the $DZ_{S_{22N}}$. Therefore, there is no CP to compute. When $i = 4, j = 1$, since $[S_{12N}]_{\text{ap}} = \{p_6, p_{10}, p_{16}, p_{17}\}$ we have $[PI_1]_{\text{map}} \subseteq [S_{12N}]_{\text{ap}}$ and $[PI_2]_{\text{map}} \subseteq [S_{12N}]_{\text{ap}}$. Thus, pCS_{12N} is constructed by merging C1 and C2 with S_{12N} . The RG of the pCS_{12N} has 13 good markings in the $LZ_{S_{12N}}$ and no states in the $DZ_{S_{12N}}$. As a result, there is no CP to compute.

Table 3: The summary of Step 10 applied for the S^3 PMR net

i, j	Subnet	Included CP	# S in net			Computed CP	# S in live controlled net	
			RG	LZ	DZ		RG	UR*
2, 1	S_{14N}	–	6	5	1	C1	5	0
2, 2	S_{15N}	–	4	3	1	C2	3	0
2, 3	S_{18N}	–	9	8	1	C3	8	0
2, 4	S_{24N}	–	4	3	1	C4	3	0
3, 1	S_{10N}	–	9	7	2	C5, C6	7	0
3, 2	S_{22N}	C1, C4	5	5	0	–	5	0
4, 1	S_{12N}	C1, C2	13	13	0	–	13	0
4, 2	S_{13N}	C1, C3	31	30	1	C7	30	0
4, 3	S_{23N}	C2, C4	8	8	0	–	8	0
5, 1	S_{8N}	C4, C5, C6	15	15	0	–	15	0
5, 2	S_{9N}	C5, C6	48	44	4	C8, C9	44	0
5, 3	S_{20N}	C1, C2, C4	12	12	0	–	12	0

(Continued)

Table 3 (continued)

i, j	Subnet	Included CP	# S in net			Computed CP	# S in live controlled net	
			RG	LZ	DZ		RG	UR*
5, 4	S_{21N}	C1, C3, C4, C7	35	35	0	–	35	0
6, 1	S_{6N}	C1, C4, C5, C6	23	23	0	–	23	0
6, 2	S_{11N}	C1, C2, C3, C7	75	75	0	–	75	0
7, 1	S_{4N}	C4, C5, C6, C8, C9	93	92	1	C10	92	0
7, 2	S_{7N}	C2, C4, C5, C6	38	38	0	–	38	0
7, 3	S_{17N}	C1, C3, C4, C7, C10	145	143	2	C11	143	0
7, 4	S_{19N}	C1, C2, C3, C4, C7	82	80	2	C12, C13	80	0
8, 1	S_{5N}	C1, C2, C4, C5, C6	54	54	0	–	54	0
9, 1	S_{2N}	C1, C3, C4, ..., C11	352	352	0	–	352	0
9, 2	S_{3N}	C2, C4, C5, C6, C8, C9, C10, C11	233	233	0	–	233	0
9, 3	S_{16N}	C1, C2, C3, C4, C7, C10, C11, C12, C13	314	314	0	–	314	0
11, 1	S_{1N}	C1, C2, ..., C13	790	790	0	–	790	0

*UR: The number of unreachable legal markings.

When $i = 4, j = 2$, as $[S_{13N}]_{ap} = \{p_{10}, p_{12}, p_{15}, p_{16}\}$ we have $[PI_1]_{map} \subseteq [S_{13N}]_{ap}$ and $[PI_3]_{map} \subseteq [S_{13N}]_{ap}$. Therefore, the partially controlled subnet S_{13N} , i.e., pCS_{13N} , is constructed by merging C1 and C3 with S_{13N} . The RG of the pCS_{13N} has 31 states, with 30 good states in the $LZ_{pCS_{13N}}$ and BM_7 in the $DZ_{pCS_{13N}}$. The BM_7 is obtained as follows: $BM_7 = p_{10} + p_{12} + 2p_{15} + p_{16}$. To stop BM_7 from being reached $PI_7 = \mu_{10} + \mu_{12} + \mu_{15} + \mu_{16} \leq 4$ is established with $[PI_7]_{map} = \{p_{10}, p_{12}, p_{15}, p_{16}\}$. The CP C7 is computed to enforce PI_7 . The controlled S_{13N} , constructed by merging the CP C7 with the pCS_{13N} , is live with 30 good markings. The remaining steps are carried out in the same manner as shown in Table 3. Table 4 depicts the PIs obtained for subnets and their marked activity places. After the completion of Step 10, in total, thirteen CPs are computed for the S^3PMR net as depicted in Table 5.

Table 4: The PIs obtained for subnets and their marked activity places

i	PI_i	Marked activity places											CP
		p2	p3	p6	p7	p8	p9	p10	p12	p15	p16	p17	
1	$\mu_{10} + \mu_{16} \leq 2$							1			1		C1
2	$\mu_6 + \mu_{17} \leq 1$			1								1	C2
3	$\mu_{12} + \mu_{15} \leq 3$								1	1			C3
4	$\mu_7 + \mu_{16} \leq 1$				1						1		C4
5	$\mu_2 + \mu_3 \leq 2$	1	1										C5
6	$\mu_3 + \mu_8 \leq 2$		1			1							C6
7	$\mu_{10} + \mu_{12} + \mu_{15} + \mu_{16} \leq 4$							1	1	1	1		C7
8	$\mu_2 + \mu_3 + \mu_9 + \mu_{15} \leq 4$	1	1				1			1			C8
9	$\mu_3 + \mu_8 + \mu_9 + \mu_{15} \leq 4$		1			1	1			1			C9
10	$\mu_7 + \mu_9 + \mu_{15} \leq 4$				1		1			1			C10

(Continued)

Table 4 (continued)

<i>i</i>	PI_i	Marked activity places											CP
		p2	p3	p6	p7	p8	p9	p10	p12	p15	p16	p17	
11	$\mu_8 + \mu_9 + \mu_{15} + \mu_{16} \leq 5$					1	1			1	1		C11
12	$\mu_6 + \mu_{10} + \mu_{15} + \mu_{16} \leq 4$			1				1		1	1		C12
13	$\mu_6 + \mu_{12} + \mu_{15} + \mu_{16} \leq 4$			1					1	1	1		C13

Table 5: CPs computed for the S^3 PMR net

<i>i</i>	PI_i	$\cdot C_i$	C_i	$\mu_0(C_i)$
1	$\mu_{10} + \mu_{16} \leq 2$	t12, t17	t10, t16	2
2	$\mu_6 + \mu_{17} \leq 1$	t6, t10, t19	t5, t17	1
3	$\mu_{12} + \mu_{15} \leq 3$	t13, t16	t12, t15	3
4	$\mu_7 + \mu_{16} \leq 1$	t7, t17	t6, t16	1
5	$\mu_2 + \mu_3 \leq 2$	t3	t1	2
6	$\mu_3 + \mu_8 \leq 2$	t3, t8	t2, t7	2
7	$\mu_{10} + \mu_{12} + \mu_{15} + \mu_{16} \leq 4$	t13, t17	t10, t15	5
8	$\mu_2 + \mu_3 + \mu_9 + \mu_{15} \leq 4$	t3, t9, t16	t1, t8, t15	4
9	$\mu_3 + \mu_8 + \mu_9 + \mu_{15} \leq 4$	t3, t9, t16	t2, t7, t15	4
10	$\mu_7 + \mu_9 + \mu_{15} \leq 4$	t7, t9, t16	t6, t8, t15	4
11	$\mu_8 + \mu_9 + \mu_{15} + \mu_{16} \leq 5$	t9, t17	t7, t15	5
12	$\mu_6 + \mu_{10} + \mu_{15} + \mu_{16} \leq 4$	t6, t12, t17	t5, t15	4
13	$\mu_6 + \mu_{12} + \mu_{15} + \mu_{16} \leq 4$	t6, t10, t13, t17	t5, t12, t15	4

Note: Redundant CPs C1 and C11 are shown in red color.

Step 11. The redundancy test carried out by using the reduced S^3 PMR net and all computed CPs $\psi = \{C1, C2, \dots, C13\}$ shows that two CPs, namely C1 and C11 are redundant while eleven CPs, namely C2, $\dots$, C10, C12, and C13, are necessary.

Step 12. The CPNM (N_c, M_c) is constructed by merging eleven necessary CPs $\psi = \{C2, \dots, C10, C12, C13\}$ with the S^3 PMR net (N, M_0) . The CPNM is live, optimal and has 3581 good states with 13,267 transitions.

Output: The live CPNM (N_c, M_c) , consisting of (N, M_0) depicted in Fig. 6 and the set of necessary CPs $\psi = \{C2, \dots, C10, C12, C13\}$ shown in Table 5.

4.2 S^3 PR Net Example 1

Fig. 9 shows a large-sized S^3 PR net of an FMS with deadlocks, consisting of 47 places, $P = \{p1-p47\}$ and thirty-eight transitions, $T = \{t1-t38\}$ from [43]. The RG of this S^3 PR net has 142,865,280 states and 779,688,927 transitions. The LZ and the DZ have 84,489,428 legal states and 58,375,852 illegal states, respectively. Consequently, an optimal live solution should be obtained with 84,489,428 legal markings.

For brevity, the details of the applied procedure to obtain an LES for this problem are omitted. By using INA, 70 siphons (SMS) of the reduced S^3 PR net are computed. Then the complementary set of siphons $[S] = \{[S_1], [S_2], \dots, [S_{70}]\}$ are computed. Finally, by applying the proposed SbDaC policy, 70 subnets are obtained, and 54 necessary CPs $\psi = \{C1, C2, \dots, C54\}$ are computed for the S^3 PR net as depicted in Table 6.

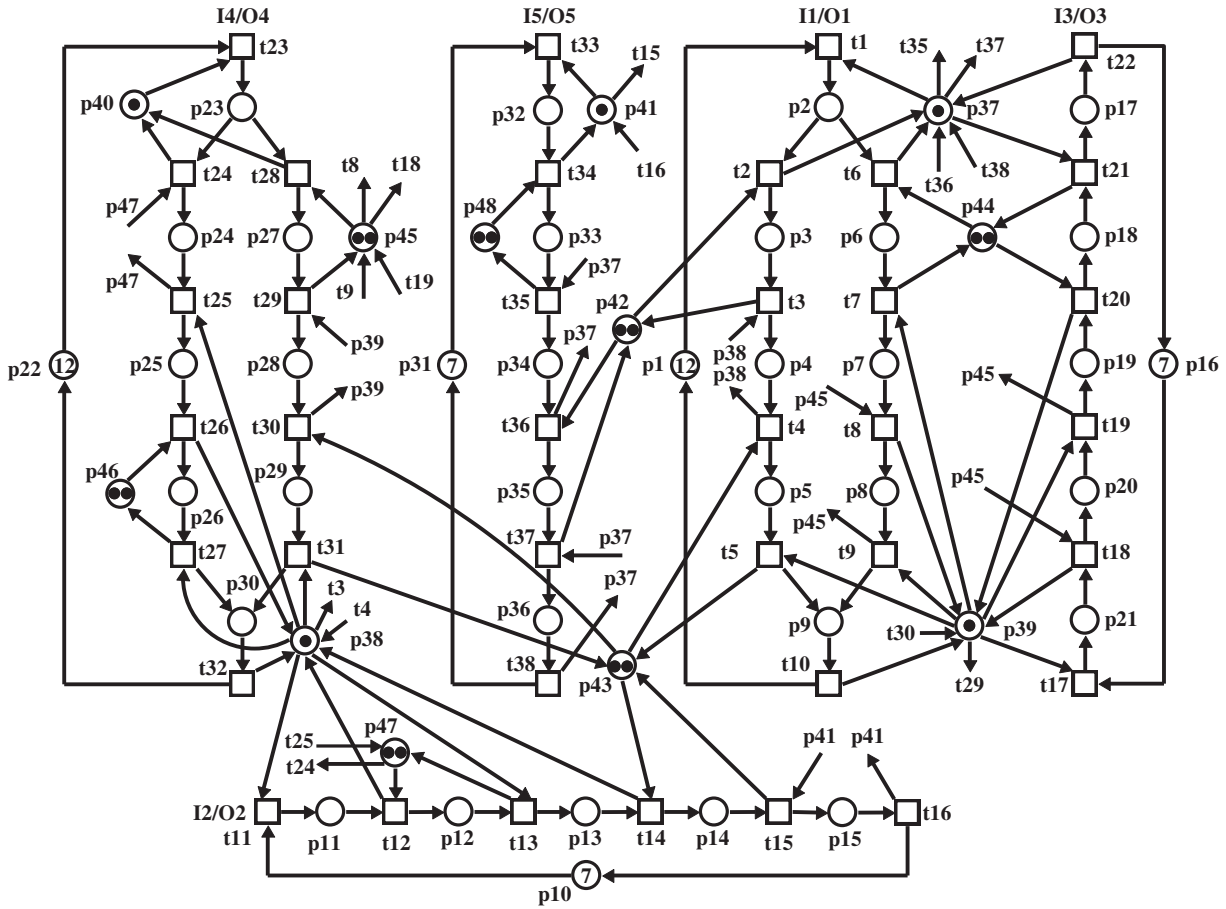


Figure 9: A large-sized S³PR net from [43]

Table 6: 54 necessary CPs computed for the large-sized S³PR net

i	PI_i	C_i	C_i	$\mu_0(C_i)$
1	$\mu_{34} + \mu_{35} \leq 2$	t37	t35	2
2	$\mu_{25} + \mu_{26} \leq 2$	t27	t25	2
3	$\mu_6 + \mu_{19} \leq 2$	t7, t20	t6, t19	2
4	$\mu_5 + \mu_{28} \leq 2$	t5, t30	t4, t29	2
5	$\mu_{11} + \mu_{12} + \mu_{24} \leq 2$	t13, t25	t11, t24	2
6	$\mu_{13} + \mu_{29} \leq 2$	t14, t31	t13, t30	2
7	$\mu_4 + \mu_{29} \leq 2$	t4, t31	t3, t30	2
8	$\mu_2 + \mu_{18} + \mu_{35} \leq 4$	t2, t6, t21, t37	t1, t20, t36	4
9	$\mu_7 + \mu_8 + \mu_{20} \leq 2$	t9, t19	t7, t18	2
10	$\mu_7 + \mu_{20} + \mu_{27} \leq 2$	t8, t19, t29	t7, t18, t28	2
11	$\mu_7 + \mu_8 + \mu_{27} \leq 2$	t9, t29	t7, t28	2
12	$\mu_{20} + \mu_{21} + \mu_{27} \leq 2$	t19, t29	t17, t28	7
13	$\mu_8 + \mu_{21} + \mu_{27} \leq 2$	t9, t18, t29	t8, t17, t28	2
14	$\mu_8 + \mu_{20} + \mu_{21} \leq 2$	t9, t19	t8, t17	2

(Continued)

Table 6 (continued)

<i>i</i>	PI_i	C_i	C_i	$\mu_0(C_i)$
15	$\mu_5 + \mu_{13} + \mu_{28} + \mu_{29} \leq 3$	t5, t14, t31	t3, t13, t29	3
16	$\mu_2 + \mu_6 + \mu_{18} + \mu_{19} + \mu_{35} \leq 5$	t2, t7, t21, t37	t1, t19, t36	5
17	$\mu_6 + \mu_{20} + \mu_{21} \leq 3$	t7, t19	t6, t17	3
18	$\mu_3 + \mu_{13} + \mu_{14} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{34} + \mu_{35} \leq 8$	t3, t15, t31, t37	t2, t13, t30, t33	8
19	$\mu_3 + \mu_4 + \mu_{14} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{34} + \mu_{35} \leq 8$	t4, t15, t31, t37	t2, t14, t30, t33	8
20	$\mu_2 + \mu_6 + \mu_{20} + \mu_{21} + \mu_{35} \leq 6$	t2, t7, t19, t21, t37	t1, t17, t20, t36	6
21	$\mu_2 + \mu_3 + \mu_4 + \mu_{14} + \mu_{18} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 10$	t4, t6, t15, t21, t31, t35, t37	t1, t14, t20, t30, t33, t36	10
22	$\mu_2 + \mu_3 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 10$	t3, t6, t15, t21, t31, t35, t37	t1, t13, t20, t30, t33, t36	10
23	$\mu_2 + \mu_5 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{28} + \mu_{32} + \mu_{33} + \mu_{35} \leq 10$	t2, t5, t7, t15, t21, t30, t35, t37	t1, t4, t14, t20, t29, t33, t36	10
24	$\mu_3 + \mu_4 + \mu_5 + \mu_{14} + \mu_{28} + \mu_{32} + \mu_{33} + \mu_{34} + \mu_{35} \leq 9$	t5, t15, t30, t37	t2, t14, t29, t33	9
25	$\mu_3 + \mu_5 + \mu_{13} + \mu_{14} + \mu_{28} + \mu_{32} + \mu_{33} + \mu_{34} + \mu_{35} \leq 9$	t3, t5, t15, t30, t37	t2, t4, t13, t29, t33	9
26	$\mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_6 + \mu_{18} + \mu_{19} + \mu_{29} + \mu_{35} \leq 8$	t5, t7, t21, t31, t37	t1, t19, t30, t36	8
27	$\mu_2 + \mu_3 + \mu_5 + \mu_6 + \mu_{13} + \mu_{18} + \mu_{19} + \mu_{29} + \mu_{35} \leq 8$	t3, t5, t7, t14, t21, t31, t37	t1, t4, t13, t19, t30, t36	8
28	$\mu_3 + \mu_5 + \mu_6 + \mu_{13} + \mu_{18} + \mu_{19} + \mu_{29} + \mu_{34} + \mu_{35} \leq 8$	t3, t5, t7, t14, t21, t31, t37	t2, t4, t6, t13, t19, t30, t35	8
29	$\mu_3 + \mu_4 + \mu_5 + \mu_6 + \mu_{18} + \mu_{19} + \mu_{29} + \mu_{34} + \mu_{35} \leq 8$	t5, t7, t21, t31, t37	t2, t6, t19, t30, t35	8
30	$\mu_2 + \mu_3 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{19} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t3, t7, t15, t21, t31, t35, t37	t1, t13, t19, t30, t33, t36	11
31	$\mu_2 + \mu_3 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{28} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t3, t7, t15, t21, t31, t35, t37	t1, t13, t20, t29, t33, t36	11
32	$\mu_2 + \mu_3 + \mu_4 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{19} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t4, t7, t15, t21, t31, t35, t37	t1, t14, t19, t30, t33, t36	11
33	$\mu_2 + \mu_3 + \mu_4 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{28} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t4, t7, t15, t21, t31, t35, t37	t1, t14, t20, t29, t33, t36	11
34	$\mu_2 + \mu_3 + \mu_5 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{19} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t3, t5, t7, t15, t21, t35, t37	t1, t4, t13, t19, t33, t36	11
35	$\mu_2 + \mu_3 + \mu_5 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{28} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t3, t5, t7, t15, t21, t30, t35, t37	t1, t4, t13, t20, t29, t33, t36	11
36	$\mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{28} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t5, t7, t15, t21, t30, t35, t37	t1, t14, t20, t29, t33, t36	11

(Continued)

Table 6 (continued)

i	PI_i	C_i	C_i	$\mu_0(C_i)$
37	$\mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{19} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t5, t7, t15, t21, t35, t37	t1, t14, t19, t33, t36	11
38	$\mu_3 + \mu_5 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{19} + \mu_{32} + \mu_{33} + \mu_{34} + \mu_{35} \leq 11$	t3, t5, t7, t15, t21, t37	t2, t4, t6, t13, t19, t33	11
39	$\mu_3 + \mu_4 + \mu_5 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{19} + \mu_{32} + \mu_{33} + \mu_{34} + \mu_{35} \leq 11$	t5, t7, t15, t21, t37	t2, t6, t14, t19, t33	11
40	$\mu_2 + \mu_6 + \mu_{14} + \mu_{19} + \mu_{20} + \mu_{27} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t2, t7, t15, t20, t29, t35, t37	t1, t14, t18, t28, t33, t36	11
41	$\mu_2 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{21} + \mu_{27} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t2, t7, t15, t18, t21, t29, t35, t37	t1, t14, t17, t20, t28, t33, t36	11
42	$\mu_2 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{20} + \mu_{27} + \mu_{32} + \mu_{33} + \mu_{35} \leq 11$	t2, t7, t15, t19, t21, t29, t35, t37	t1, t14, t18, t20, t28, t33, t36	11
43	$\mu_2 + \mu_3 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{21} + \mu_{27} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 12$	t3, t7, t15, t18, t21, t29, t31, t35, t37	t1, t13, t17, t20, t28, t30, t33, t36	12
44	$\mu_2 + \mu_3 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{20} + \mu_{27} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 12$	t3, t7, t15, t19, t21, t29, t31, t35, t37	t1, t13, t18, t20, t28, t30, t33, t36	12
45	$\mu_2 + \mu_3 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{19} + \mu_{20} + \mu_{27} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 12$	t3, t7, t15, t20, t29, t31, t35, t37	t1, t13, t18, t28, t30, t33, t36	12
46	$\mu_2 + \mu_3 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{20} + \mu_{21} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 12$	t3, t7, t15, t19, t21, t31, t35, t37	t1, t13, t17, t20, t30, t33, t36	12
47	$\mu_2 + \mu_3 + \mu_4 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{21} + \mu_{27} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 12$	t4, t7, t15, t18, t21, t29, t31, t35, t37	t1, t14, t17, t20, t28, t30, t33, t36	12
48	$\mu_2 + \mu_3 + \mu_4 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{20} + \mu_{27} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 12$	t4, t7, t15, t19, t21, t29, t31, t35, t37	t1, t14, t18, t20, t28, t30, t33, t36	12
49	$\mu_2 + \mu_3 + \mu_4 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{20} + \mu_{21} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 12$	t4, t7, t15, t19, t21, t31, t35, t37	t1, t14, t17, t20, t30, t33, t36	12
50	$\mu_2 + \mu_3 + \mu_4 + \mu_6 + \mu_{14} + \mu_{19} + \mu_{20} + \mu_{27} + \mu_{29} + \mu_{32} + \mu_{33} + \mu_{35} \leq 12$	t4, t7, t15, t20, t29, t31, t35, t37	t1, t14, t18, t28, t30, t33, t36	12
51	$\mu_2 + \mu_4 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{20} + \mu_{27} + \mu_{28} + \mu_{32} + \mu_{33} + \mu_{35} \leq 12$	t2, t4, t7, t15, t19, t21, t30, t35, t37	t1, t3, t14, t18, t20, t28, t33, t36	12
52	$\mu_2 + \mu_3 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{20} + \mu_{27} + \mu_{28} + \mu_{32} + \mu_{33} + \mu_{35} \leq 12$	t3, t7, t15, t19, t21, t30, t35, t37	t1, t13, t18, t20, t28, t33, t36	12
53	$\mu_4 + \mu_6 + \mu_{14} + \mu_{18} + \mu_{20} + \mu_{27} + \mu_{28} + \mu_{32} + \mu_{33} + \mu_{34} + \mu_{35} \leq 12$	t4, t7, t15, t19, t21, t30, t37	t3, t6, t14, t18, t20, t28, t33	12

(Continued)

Table 6 (continued)

i	PI_i	C_i	C_i	$\mu_0(C_i)$
54	$\mu_3 + \mu_6 + \mu_{13} + \mu_{14} + \mu_{18} + \mu_{20} +$ $\mu_{27} + \mu_{28} + \mu_{32} + \mu_{33} + \mu_{34} + \mu_{35}$ ≤ 12	t3, t7, t15, t19, t21, t30, t37	t2, t6, t13, t18, t20, t28, t33	12

The CPNM (N_c, M_c) is constructed by merging 54 necessary CPs with the S^3PR net (N, M_0) shown in Fig. 9. The CPNM is live with 84,480,488 good markings. The permissiveness of the CPNM is $84,480,488/84,489,428 = 99.989\%$.

Table 7 depicts the comparison of different control methods for the large-sized S^3PR net based on behavioral permissiveness and structural complexity. It is well-known that a low structural complexity of a given LES means fewer numbers and simpler structure of CPs [52,53], which can provide low implementation overheads in terms of both software and hardware costs. Therefore, from this perspective, fewer numbers of CPs with ordinary arcs is desirable. Another aspect in assessing the LESs is that of behavioral permissiveness. The LES provided in [54] consists of only 10 CPs with ordinary arcs but its behavioral permissiveness is very low. The LESs of both [23,27] provide maximally permissiveness, but they contain CPs with weighted arcs. Although solutions obtained by methods [23,27] provide optimal permissive behavior, these methods require solving ILPPs, which are intractable for very large-scale nets. The LES shown in Table 6 provides a near-optimal solution with a simpler structure of CPs having only ordinary arcs.

Table 7: Comparison of different control methods for the large-sized S^3PR net

Parameters	Solution provided in [54]	Solution provided in [23]	Solution provided in [27]	Solution provided in Table 6
# Total CPs added	10	25	53	54
# CPs with ordinary arcs	10	17	51	54
# CPs with weighted arcs	–	8	2	–
# reachable states	257 890	84 489 428	84 489 428	84 480 488
Permissiveness (%)	$257,890/84,489,428 =$ 0.30	$84,489,428/84,489,428$ = 100	$84,489,428/84$ $489,428 = 100$	$84,480,488/84,489,428$ = 99.989

Note: Even though 25 CPs indicated in Column 3 were computed in [23], they are made available in [43].

4.3 S^3PR Net Example 2

Fig. 10 shows another large-sized S^3PR net of an FMS with deadlocks, consisting of 26 places, $P = \{p1-p26\}$ and twenty transitions, $T = \{t1-t20\}$ from [55]. The RG of this S^3PR net has 74,202,530 states. The LZ and the DZ have 59,698,137 legal states and 14,504,393 illegal states, respectively. Consequently, an optimal live solution should be obtained with 59,698,137 legal markings.

The details of the applied procedure for this problem are omitted. By using INA, 18 SMS $S = \{S_1, S_2, \dots, S_{18}\}$ of the reduced S^3PR net are computed. Then the complementary set of siphons $[S] = \{[S_1], [S_2], \dots, [S_{18}]\}$ are computed. Finally, by applying the proposed SbDaC policy, 18 subnets are obtained and 17 necessary CPs $\psi = \{C1, \dots, C17\}$ are computed for the S^3PR net as depicted in Table 8.

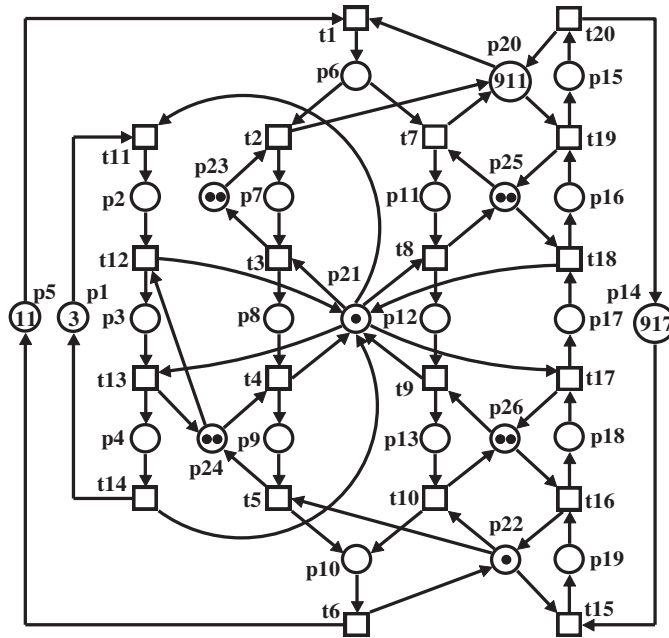


Figure 10: A large-sized S³PR net from [55]

Table 8: 17 necessary CPs computed for the second large-sized S³PR net

i	PI_i	$\bullet C_i$	C_r	$\mu_0(C_i)$
1	$\mu_{13} + \mu_{19} \leq 2$	t10, t16	t9, t15	2
2	$\mu_2 + \mu_3 \leq 2$	t13	t11	2
3	$\mu_3 + \mu_8 \leq 2$	t4, t13	t3, t12	2
4	$\mu_{12} + \mu_{18} \leq 2$	t9, t17	t8, t16	2
5	$\mu_{11} + \mu_{17} \leq 2$	t8, t18	t7, t17	2
6	$\mu_{11} + \mu_{18} \leq 3$	t8, t17	t7, t16	3
7	$\mu_{12} + \mu_{13} + \mu_{18} + \mu_{19} \leq 3$	t10, t17	t8, t15	3
8	$\mu_{11} + \mu_{12} + \mu_{18} + \mu_{19} \leq 4$	t9, t17	t7, t15	4
9	$\mu_{11} + \mu_{13} + \mu_{18} + \mu_{19} \leq 4$	t8, t10, t17	t7, t9, t15	4
10	$\mu_2 + \mu_3 + \mu_9 + \mu_{13} + \mu_{18} + \mu_{19} \leq 5$	t5, t10, t13, t17	t4, t9, t11, t15	5
11	$\mu_3 + \mu_8 + \mu_9 + \mu_{13} + \mu_{18} + \mu_{19} \leq 5$	t5, t10, t13, t17	t3, t9, t12, t15	5
12	$\mu_2 + \mu_3 + \mu_9 + \mu_{11} + \mu_{18} + \mu_{19} \leq 6$	t5, t8, t13, t17	t4, t7, t11, t15	6
13	$\mu_3 + \mu_8 + \mu_9 + \mu_{11} + \mu_{18} + \mu_{19} \leq 6$	t5, t8, t13, t17	t3, t7, t12, t15	6
14	$\mu_6 + \mu_7 + \mu_{11} + \mu_{16} + \mu_{17} \leq 915$	t3, t8, t19	t1, t17	915
15	$\mu_6 + \mu_7 + \mu_8 + \mu_9 + \mu_{11} + \mu_{13} + \mu_{16} + \mu_{18} + \mu_{19} \leq 919$	t5, t8, t10, t17, t19	t1, t9, t15, t18	919
16	$\mu_6 + \mu_7 + \mu_9 + \mu_{11} + \mu_{13} + \mu_{17} + \mu_{18} + \mu_{19} \leq 919$	t3, t5, t8, t10, t18	t1, t4, t9, t15	919
17	$\mu_6 + \mu_7 + \mu_9 + \mu_{11} + \mu_{12} + \mu_{16} + \mu_{18} + \mu_{19} \leq 919$	t3, t5, t9, t17, t19	t1, t4, t15, t18	919

The CPNM (N_c , M_c) is constructed by merging 17 CPs with the S³PR net (N , M_0) shown in Fig. 10. The CPNM is live with 59,698,137 legal markings. The permissiveness of the CPNM is 100%.

Table 9 depicts the comparison of two control policies for the second large-sized S³PR. Although the LES provided in [55] consists of only 4 CPs with weighted arcs, its behavioral permissiveness is 98,88%. On the other hand, the LES provided in Table 8 provides maximally permissive controlled behavior with 17 CPs having only ordinary arcs.

Table 9: Comparison of two policies for the second large-sized S³PR net

Parameters	Solution provided in [55]	Solution provided in Table 8
# Total CPs added	4	17
# CPs with ordinary arcs	–	17
# CPs with weighted arcs	4	–
# reachable states	59 030 159	59 698 137
Permissiveness (%)	59,030,159/59,698,137 = 98.88	59,698,137/59,698,137 = 100

4.4 S^3PGR^2 Net Example

Fig. 11 shows a large-sized S³PGR² net of an FMS with deadlock problems, which has $P = \{p1-p26\}$ and $T = \{t1-t22\}$ from [56]. Originally the markings of idle places $p1$, $p7$, and $p16$ were $M(p1) = M(p7) = M(p16) = 3$. The RG of this PNM has 170,312,521 markings and 1,428,308,870 transitions. The LZ and the DZ have 168,026,565 legal states and 2,285,956 illegal states, respectively. Therefore, an optimal live solution must provide 168,026,565 legal markings.

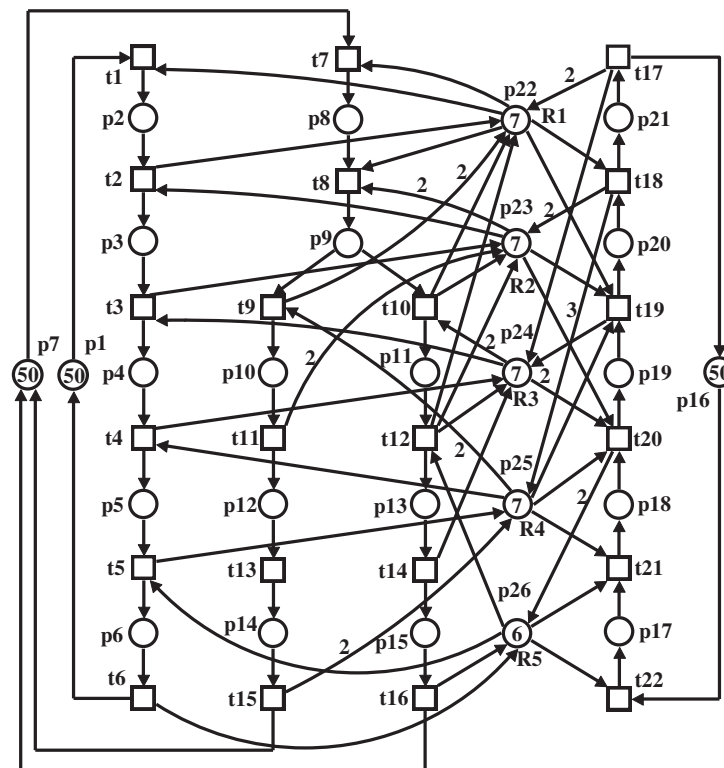


Figure 11: A large-sized S^3PGR^2 net from [56] with $M(p1) = M(p7) = M(p16) = 50$

The details of the applied procedure for this problem are omitted. By using INA, 19 SMSs of the reduced S^3PGR^2 net are computed. Then the complementary set of siphons $[S] = \{[S_1], [S_2], \dots, [S_{19}]\}$ are computed. Finally, by applying the proposed SbDaC policy, 19 subnets are obtained and 26 necessary CPs $\psi = \{C_1, \dots, C_{26}\}$ with only ordinary arcs are synthesized for the S^3PGR^2 net as depicted in Table 10. The CPNM (N_c, M_c) is constructed by merging 26 CPs with the S^3PGR^2 net (N, M_0) depicted in Fig. 11. The CPNM is live with 78,842,738 good markings. The permissiveness of the CPNM is $78,842,738/168,026,565 = 46.92\%$.

Table 10: Computed 26 necessary CPs for the S^3PGR^2 net

i	PI_i	$\bullet C_i$	C_i	$\mu_0(C_i)$
1	$\mu_{18} + \mu_{19} \leq 3$	t19	t21	3
2	$\mu_8 + \mu_{20} \leq 6$	t8, t18	t7, t19	9 6
3	$\mu_{11} + \mu_{17} + \mu_{18} \leq 6$	t12, t20	t10, t22	6 5
4	$\mu_5 + \mu_{18} \leq 6$	t5, t20	t4, t21	6
5	$\mu_5 + \mu_{17} + \mu_{18} + \mu_{19} \leq 7$	t5, t19	t4, t22	7
6	$\mu_8 + \mu_9 + \mu_{19} \leq 6$	t9, t10, t19	t7, t20	6
7	$\mu_4 + \mu_9 + \mu_{19} + \mu_{20} \leq 6$	t4, t9, t10, t18	t3, t8, t20	6
8	$\mu_8 + \mu_9 + \mu_{11} + \mu_{17} + \mu_{18} + \mu_{19} + \mu_{20} \leq 9$	t9, t12, t18	t7, t22	9
9	$\mu_2 + \mu_8 + \mu_{11} + \mu_{18} + \mu_{20} \leq 9$	t2, t8, t12, t18, t20	t1, t7, t10, t19, t21	9
10	$\mu_9 + \mu_{19} \leq 4$	t9, t10, t19	t8, t20	4
11	$\mu_2 + \mu_3 + \mu_4 + \mu_8 + \mu_9 + \mu_{19} + \mu_{20} \leq 9$	t4, t9, t10, t18	t1, t7, t20	9
12	$\mu_4 + \mu_5 + \mu_{11} + \mu_{18} \leq 9$	t5, t12, t20	t3, t10, t21	9
13	$\mu_4 + \mu_8 + \mu_9 + \mu_{11} + \mu_{18} + \mu_{20} \leq 9$	t4, t9, t12, t18, t20	t3, t7, t19, t21	9
14	$\mu_5 + \mu_8 + \mu_9 + \mu_{11} + \mu_{17} + \mu_{18} + \mu_{20} \leq 9$	t5, t9, t12, t18, t20	t4, t7, t19, t22	9
15	$\mu_2 + \mu_5 + \mu_9 + \mu_{11} + \mu_{17} + \mu_{18} + \mu_{20} \leq 9$	t2, t5, t9, t12, t18, t20	t1, t4, t8, t19, t22	9
16	$\mu_3 + \mu_8 + \mu_9 + \mu_{11} + \mu_{18} + \mu_{20} \leq 9$	t3, t9, t12, t18, t20	t2, t7, t19, t21	9
17	$\mu_3 + \mu_4 + \mu_9 + \mu_{17} + \mu_{19} \leq 10$	t4, t9, t10, t19, t21	t2, t8, t20, t22	10
18	$\mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_8 + \mu_9 + \mu_{11} + \mu_{18} + \mu_{20} \leq 11$	t5, t9, t12, t18, t20	t1, t7, t19, t21	11
19	$\mu_2 + \mu_3 + \mu_4 + \mu_8 + \mu_9 + \mu_{11} + \mu_{17} + \mu_{18} + \mu_{19} + \mu_{20} \leq 13$	t4, t9, t12, t18	t1, t7, t22	13
20	$\mu_2 + \mu_3 + \mu_4 + \mu_9 + \mu_{11} + \mu_{18} + \mu_{20} \leq 10$	t4, t9, t12, t18, t20	t1, t8, t19, t21	10
21	$\mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_{11} + \mu_{17} + \mu_{18} + \mu_{19} \leq 11$	t5, t12, t19	t1, t10, t22	11
22	$\mu_4 + \mu_5 + \mu_8 + \mu_9 + \mu_{11} + \mu_{17} + \mu_{18} + \mu_{19} \leq 11$	t5, t9, t12, t19	t3, t7, t22	11
23	$\mu_2 + \mu_3 + \mu_4 + \mu_8 + \mu_{17} + \mu_{18} + \mu_{20} \leq 11$	t4, t8, t18, t20	t1, t7, t19, t22	11
24	$\mu_3 + \mu_4 + \mu_5 + \mu_9 + \mu_{11} + \mu_{17} + \mu_{18} + \mu_{19} \leq 11$	t5, t9, t12, t19	t2, t8, t22	11
25	$\mu_2 + \mu_4 + \mu_5 + \mu_9 + \mu_{11} + \mu_{17} + \mu_{18} + \mu_{19} \leq 11$	t2, t5, t9, t12, t19	t1, t3, t8, t22	11
26	$\mu_3 + \mu_4 + \mu_8 + \mu_{17} + \mu_{18} + \mu_{19} \leq 12$	t4, t8, t19	t2, t7, t22	12

5 Conclusion

A novel siphon-based divide-and-conquer (SbDaC) policy is presented in this paper for the synthesis of PN-based LEs for FMSs suffering from deadlocks or livelocks. Given the uncontrolled and bounded PN model of an FMS with deadlock or livelock problems, a live controlled PN model of the FMS can be obtained using the SbDaC method proposed in this paper. The proposed SbDaC policy is applicable to all classes of PNs related to FMS having deadlocks or livelocks. Therefore, it is general and its use is not limited to any

subclass of PNs of FMS. In theory, its off-line computation is of exponential complexity, but it is generally applicable, easy to use, effective and straightforward. The applicability and the effectiveness of the proposed method to realistic systems have been shown by considering several examples from the literature. Especially, three of these examples contain realistic FMSs with very large state spaces. In general, the method provides an optimal or a near-optimal solution in the existence of the optimal one. However, for some generalized classes of PN models of FMS, some legal states cannot be reached under the controlled behavior, resulting in less permissive behavior. Therefore, further research is necessary to provide new policies to achieve more permissive LESs for generalized classes of PNs. It is also necessary to carry out further research to reduce the structural complexity of the computed LESs.

Acknowledgement: The authors extend their appreciation to King Saud University, Saudi Arabia for funding this work through the Ongoing Research Funding Program (ORF-2025-704), King Saud University, Riyadh, Saudi Arabia.

Funding Statement: The authors extend their appreciation to King Saud University, Saudi Arabia for funding this work through the Ongoing Research Funding Program (ORF-2025-704), King Saud University, Riyadh, Saudi Arabia.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization: Murat Uzam, Wei Wei, Yufeng Chen; methodology: Murat Uzam, Wei Wei, Yufeng Chen, Mohammed El-Meligy; software: Bernard Berthomieu, Wei Wei, Yufeng Chen, Mohamed Abdel Fattah Sharaf; validation: Bernard Berthomieu, Mohammed El-Meligy, Mohamed Abdel Fattah Sharaf; formal analysis: Murat Uzam, Wei Wei, Yufeng Chen, Mohammed El-Meligy; analysis and interpretation of results: Murat Uzam, Bernard Berthomieu, Yufeng Chen; draft manuscript preparation: Murat Uzam, Bernard Berthomieu, Yufeng Chen, Mohamed Abdel Fattah Sharaf; review and editing: Bernard Berthomieu, Mohammed El-Meligy, Mohamed Abdel Fattah Sharaf; funding acquisition: Mohammed El-Meligy, Mohamed Abdel Fattah Sharaf. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Petri net definitions of the example Petri nets utilized in this article are provided for the analysis of these nets by using INA and/or TINA Petri net tools.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Li Z, Zhou M. Deadlock resolution in automated manufacturing systems: a novel petri net approach. Berlin/Heidelberg, Germany: Springer; 2009. doi:10.1007/978-1-84882-244-3.
2. David R, Alla H. Discrete, continuous, and hybrid petri nets. Berlin/Heidelberg, Germany: Springer; 2005. doi:10.1007/978-3-642-10669-9.
3. Wu NQ, Zhou MC, Li ZW. Short-term scheduling of crude-oil operations: enhancement of crude-oil operations scheduling using a Petri net-based control-theoretic approach. IEEE Robot Autom Mag. 2015;22(2):64–76. doi:10.1109/MRA.2015.2417651.
4. Xing K, Hu B, Chen H. Deadlock avoidance policy for flexible manufacturing systems. In: Zhou MC, editor. Petri nets in flexible and agile automation. Boston, MA, USA: Kluwer; 1995. p. 239–63. doi:10.1007/978-1-4615-2231-7_9.
5. Xing K, Hu B, Chen H. Deadlock avoidance policy for Petri-net modeling of flexible manufacturing systems with shared resources. IEEE Trans Autom Control. 1996;41(2):289–95. doi:10.1109/9.481517.
6. Yang Y, Liu Z, Zhao Z, Zhou M. Deadlock analysis and avoidance for automated manufacturing systems based on petri nets with forward-conflict-free structures. IEEE Trans Syst Man Cybern Syst. 2025;55(3):1634–46. doi:10.1109/TSMC.2024.3509901.

7. Capkovic F. Modelling and control of resource allocation systems within discrete event systems by means of Petri nets—part 1: invariants, siphons and traps in deadlock avoidance. *Comput Inform.* 2021;40(3):648–89. doi:10.31577/cai_2021_3_648.
8. Yang B, Hu H. Maximally permissive deadlock and livelock avoidance for automated manufacturing systems via critical distance. *IEEE Trans Autom Sci Eng.* 2022;19(4):3838–52. doi:10.1109/TASE.2021.3138169.
9. Tat Leung Y, Sheen G-J. Resolving deadlocks in flexible manufacturing cells. *J Manuf Syst.* 1993;12(4):291–304. doi:10.1016/0278-6125(93)90320-S.
10. Cho H, Kumaran TK, Wysk RA. Graph-theoretic deadlock detection and resolution for flexible manufacturing systems. *IEEE Trans Robot Autom.* 1995;11(3):413–21. doi:10.1109/70.388784.
11. Pan YL, Tseng CY, Chen JC. Enhancement of computational efficiency for deadlock recovery of flexible manufacturing systems using improved generating and comparing aiding matrix algorithms. *Processes.* 2023;11(10):3026. doi:10.3390/pr11103026.
12. Ezpeleta J, Colom JM, Martinez J. A Petri net based deadlock prevention policy for flexible manufacturing systems. *IEEE Trans Robot Autom.* 1995;11(2):173–84. doi:10.1109/70.370500.
13. 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. doi:10.1007/s001700200014.
14. Wang S, Guo X, Karoui O, Zhou M, You D, Abusorrah A. A refined siphon-based deadlock prevention policy for a class of Petri nets. *IEEE Trans Syst Man Cybern Syst.* 2023;53(1):191–203. doi:10.1109/TSMC.2022.3174421.
15. Li C, Jin F, Chen Y, Li Z, Uzam M, Ma H. Calculation and analysis of Petri net reachability graphs by a think-globally-act locally method. *Mathematics.* 2025;13(5):793. doi:10.3390/math13050793.
16. Capkovic F. Petri net-based S^3PR models of automated manufacturing systems with resources and their deadlock prevention. *Acta Polytech Hung.* 2023;20(6):79–96.
17. Row T, Pan Y. Incomparable single controller for solving deadlock problems of flexible manufacturing systems. *IEEE Access.* 2023;11:45270–8. doi:10.1109/ACCESS.2023.3272227.
18. Yang Y, Yang J, Liang N, Zhong C. Control law for two-process flexible manufacturing systems modeled using petri nets. *Mathematics.* 2025;13(4):611. doi:10.3390/math13040611.
19. Ma T, Wang J, Hu H. A strict minimal siphon-based necessary and sufficient condition for liveness verification in general petri nets. *IEEE Access.* 2025;13:53517–30. doi:10.1109/ACCESS.2025.3551396.
20. Yang B, Hu H. On the equivalence between robustness and liveness in automated manufacturing systems. *IEEE Trans Syst Man Cybern Syst.* 2024;54(12):7495–507. doi:10.1109/TSMC.2024.3458939.
21. Fan X, Hu H, Yang B, Liu Y, He G. Event circuit structures for deadlock avoidance in flexible manufacturing systems. *IEEE Trans Autom Sci Eng.* 2023;20(1):597–610. doi:10.1109/TASE.2022.3163854.
22. Huang YS, Jeng MD, Xie X, Chung DH. Siphon-based deadlock prevention policy for flexible manufacturing systems. *IEEE Trans Syst Man Cybern Part A Syst Hum.* 2006;36(6):1248–56. doi:10.1109/TSMCA.2006.878953.
23. Piroddi L, Cordone R, Fumagalli I. Combined siphon and marking generation for deadlock prevention in Petri nets. *IEEE Trans Syst Man Cybern Part A Syst Hum.* 2009;39(3):650–61. doi:10.1109/TSMCA.2009.2013189.
24. Hou Y, Li ZW, Zhao M, Liu D. Extended elementary siphon-based deadlock prevention policy for a class of generalised Petri nets. *Int J Comput Integr Manuf.* 2014;21(1):85–102. doi:10.1080/0951192X.2013.800233.
25. Hou Y, Zhao M, Liu D, Hong L. An efficient siphon-based deadlock prevention policy for a class of generalized Petri nets. *Discret Dyn Nat Soc.* 2016;2016:8219424. doi:10.1155/2016/8219424.
26. Uzam M, Zhou MC. An iterative synthesis approach to Petri net-based deadlock prevention policy for flexible manufacturing systems. *IEEE Trans Syst Man Cybern Part A Syst Hum.* 2007;37(3):362–71. doi:10.1109/TSMCA.2007.893484.
27. 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.* 2010;8(2):374–93. doi:10.1109/TASE.2010.2060332.
28. Xing K, Zhou M, Wang F, Liu H, Tian F. Resource-transition circuits and siphons for deadlock control of automated manufacturing systems. *IEEE Trans Syst Man Cybern Part A Syst Hum.* 2011;41(1):74–84. doi:10.1109/TSMCA.2010.2048898A.
29. Capkovic F. Control of deadlocked discrete-event systems using petri nets. *Acta Polytech Hung.* 2022;19(2):213–33.

30. Al-Shayea A, Kaid H, Al-Ahmari A, Nasr EA, Kamrani AK, Mahmoud HA. Colored resource-oriented petri nets for deadlock control and reliability design of automated manufacturing systems. *IEEE Access*. 2021;9:125616–27. doi:10.1109/ACCESS.2021.3111575.
31. Bashir M, Zhou J, Muhammad BB. Optimal supervisory control for flexible manufacturing systems model with petri nets: a place-transition control. *IEEE Access*. 2021;9:58566–78. doi:10.1109/ACCESS.2021.3072892.
32. Tseng C, Chen J, Pan Y. An improved deadlock recovery policy of flexible manufacturing systems based on resource flow graphs. *IEEE Access*. 2024;12(3):65202–12. doi:10.1109/ACCESS.2024.3396879.
33. Dou H, Zhou M, Wang S, Albeshri A. An efficient liveness analysis method for petri nets via maximally good-step graphs. *IEEE Trans Syst Man Cybern Syst*. 2024;54(7):3908–19. doi:10.1109/TSMC.2024.3372941.
34. Feng Y, Ren S, Cao Y, Xing K, Yang Y. Deadlock control for flexible assembly systems with multiple resource requirements and separately-loaded parts. *IEEE Trans Autom Sci Eng*. 2025;22(2):9275–84. doi:10.1109/TASE.2024.3504714.
35. Li Y, Chen Y, Zhou R. A set covering approach to design maximally permissive supervisors for flexible manufacturing systems. *Mathematics*. 2024;12(11):1687. doi:10.3390/math12111687.
36. Uzam M, El-Sherbeeney AM, Guo W, Li ZW. Design of an improved think globally act locally approach for the computation of Petri nets based liveness-enforcing supervisors of FMSs. *IEEE Access*. 2024;12(4):74367–88. doi:10.1109/ACCESS.2024.3403804.
37. Li Z, Zhu S, Zhou M. A Divide-and-conquer strategy to deadlock prevention in flexible manufacturing systems. *IEEE Trans Syst Man Cybern*. 2009;39(2):156–69. doi:10.1109/TSMCC.2008.2007246.
38. Uzam M, Li Z, Gelen G, Zakariyya R. A divide-and-conquer-method for the synthesis of liveness-enforcing supervisors for flexible manufacturing systems. *J Intell Manuf*. 2016;27(5):1111–29. doi:10.1007/s10845-014-0938-z.
39. Zhong C, He W, Li Z, Wu N, Qu T. Deadlock analysis and control using Petri net decomposition techniques. *Inf Sci*. 2019;482(6):440–56. doi:10.1016/j.ins.2019.01.029.
40. 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. doi:10.1080/00207540500431321.
41. INA Integrated Net Analyzer, a software tool for analysis of PNs, Version 2.2 [Internet]. 2022 [cited 2025 Jul 29]. Available from: <http://www.informatik.hu-berlin.de/~starke/ina.html>.
42. TINA. Time petri net analyzer [Internet]. [cited 2025 Jul 29]. Available from: <https://projects.laas.fr/tina/index.php>.
43. Uzam M, Liu D, Berthomieu B, Gelen G, Zhang Z, Mostafa AM, et al. On deadlock/livelock studies based on reachability graph of Petri nets by using TINA. *IEEE Access*. 2024;12(3):135506–34. doi:10.1109/ACCESS.2024.3461168.
44. Liu G, Barkaoui K. A survey of siphons in Petri nets. *Inf Sci*. 2016;363:198–220. doi:10.1016/j.ins.2016.02.010.
45. Li ZW, Zhou M. Elementary siphons of Petri nets and their application to deadlock prevention in flexible manufacturing systems. *IEEE Trans Syst Man Cybern Part A Syst Hum*. 2004;34(1):38–51. doi:10.1109/TSMCA.2003.820576.
46. Chen YF, Li ZW, Al-Ahmari A. Nonpure Petri net supervisors for optimal deadlock control of flexible manufacturing systems. *IEEE Trans Syst Man Cybern Part A Syst Hum*. 2013;43(2):252–65. doi:10.1109/TSMCA.2012.2202108.
47. Yamalidou K, Moody J, Lemmon M, Antsaklis P. Feedback control of Petri nets based on place invariants. *Automatica*. 1996;32(1):15–28. doi:10.1016/0005-1098(95)00103-4.
48. Uzam M, Li Z, Zhou M. Identification and elimination of redundant control places in Petri net based liveness-enforcing supervisors of FMS. *Int J Adv Manuf Technol*. 2007;35(1–2):150–68. doi:10.1007/s00170-006-0701-5.
49. Heiner M, Schwarick M, Wegener JT. Charlie—an extensible petri net analysis tool. In: Devillers R, Valmari A, editors. In: Application and theory of petri nets and concurrency. Vol. 9115. Cham, Switzerland: Springer; 2015. doi:10.1007/978-3-319-19488-2_10.
50. Liu F, Assaf G, Chen M, Heiner M. A Petri nets-based framework for whole-cell modeling. *Biosystems*. 2021;210(0237):104533. doi:10.1016/j.biosystems.2021.104533.

51. Yan M, Li Z, Wei N, Zhao M. A deadlock prevention policy for a class of Petri nets S3PMR*. *J Inf Sci Eng.* 2009;25(1):167–83. doi:10.6688/JISE.2009.25.1.9.
52. Feng Y, Ren S, Ren X, Chen H, Yang Y. Small-size liveness-enforcing supervisor for automated manufacturing systems using the theory of transition cover. *IEEE Trans Syst Man Cybern Syst.* 2023;53(4):2222–35. doi:10.1109/TSMC.2022.3209156.
53. Row T, Tseng C, Pan Y. Universal single controller for solving deadlock problems of flexible manufacturing systems. *IEEE Access.* 2024;12(12):191907–20. doi:10.1109/ACCESS.2024.3515462.
54. Li Z, Zhou M. Comparison of two deadlock prevention methods for different-size flexible manufacturing systems. *Int J Intell Control Syst.* 2005;10(3):235–43.
55. Dou H, You D, Wang S, Zhou M. Designing liveness-enforcing supervisors for manufacturing systems by using maximally good step graphs of petri nets. *IEEE Trans Autom Sci Eng.* 2025;22:7312–23. doi:10.1109/TASE.2024.3450656.
56. Hu H, Li Z. Local and global deadlock prevention policies for resource allocation systems using partially generated reachability graphs. *Comput Ind Eng.* 2009;57(4):1168–81. doi:10.1016/j.cie.2009.05.006.