IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 24, 2021, accepted April 8, 2021, date of publication April 13, 2021, date of current version April 22, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3072892

Optimal Supervisory Control for Flexible Manufacturing Systems Model With Petri Nets: A Place-Transition Control

MUHAMMAD BASHIR[1], JIAN ZHOU[1], AND BASHIR BALA MUHAMMAD[2,3]
[1]School of Electronics and Information, Xi'an Polytechnic University, Xi'an 710048, China
[2]Department of Mechatronics Engineering, Faculty of Air Engineering, Air Force Institute of Technology, Kaduna 800282, Nigeria
[3]School of Mechanical Engineering, Air Force Institute of Technology, Kaduna 800282, Nigeria

Corresponding author: Muhammad Bashir (m.bashir71@yahoo.com)

This work was supported in part by the School of Electronic and Information, Xi'an Polytechnic University, under Project 310/107020454.

ABSTRACT Supervisory control for discrete event systems that is widely studied in the literature to tackle the Petri net model's deadlock occurrence of flexible manufacturing systems. A supervisory structure constructed using control places imposed restrictions on the concurrent operation in the systems, which degrade the supervisor's efficiency. This study proposed a new method to construct a supervisory structure using combine control places and control transitions to ensure flexible manufacturing systems' smooth operation. Place-Transition controller is designed for each concurrent process of the systems. Three algorithms are proposed in this method. The algorithm 1 computes the loop markings at each process of the Petri net model. Algorithm 2 is used to sort the deadlock markings based on the concurrent processes of the Petri net model. The Transition-Place controller and Place-Transition controller are designed using algorithm 3. Transition-Place controller creates extra states in the reachability graph, which is called dump markings. Place-Transition controller projects the deadlock markings to the loop markings via dump markings. The number of states created by the Transition-Place controller depends on the number of concurrent states in the Petri net model. The final supervisory structure controls the deadlock occurrence of flexible manufacturing systems with zero restrictions of systems operation. This proposed method is efficient as it retains all the states generated in the controlled Petri net model.

INDEX TERMS Deadlock, Petri nets, flexible manufacturing systems, supervisory control, transition control.

I. INTRODUCTION

Automated controlled for discrete event systems (DESs) are widely used in flexible manufacturing systems (FMSs) due to their applicability in controlling and boosting processed product growth. Its flexibility makes the systems to be adjusted to produce the desire specifications based on the needs. Although, deadlocks occurrence is the persistence problem that degrades the benefits derives from FMSs, and needs to be tackled. Several supervisory controlled methods for DESs are proposed in the literature to addressed the deadlock occurrence in FMSs.

A Petri net is a mathematical tool that can best describe FMSs properties, i.e., concurrency, synchronization, and conflict. Two approaches exist to analyze the Petri net model (PNM) for FMSs: (i) structural analysis and (ii) reachability graph analysis. Reachability graph analysis requires the full or partial computations of reachable markings. Structural analysis usually utilizes the PNM's structural properties such as siphon, resource transition circuit (RTC), place/transition (P/T) invariants to determine the potential deadlock markings and controlled them. Supervisor design using a reachability graph provides optimal solutions, while in most cases, the supervisory control design using structural analysis provides suboptimal solutions. Because, in the structural analysis, it is challenging to ensure all the states in the system. The supervisory control for discrete event systems can be further categorized into two approaches: (i) use of control places and (ii) use of control transitions.

Control places are used in the works of [1]–[17], [35], [36] to enforce certain constraints that satisfied a specified set of instructions, to restrict the firing of transitions that lead the

The associate editor coordinating the review of this manuscript and approving it for publication was Zhiwu Li.

systems to deadlock zone. Ghaffari developed the theory of regions in [19] for optimal supervisors using control places to realize the desired behavior. Too many inequality constraints are generated to ensure a permissive supervisor, which complicates the supervisory structure's computation. An iterative synthesis supervisory control for the PNM of FMSs is developed in [20]. The method classified the reachability markings into two regions: live zone (LZ) and deadlock zone (DZ). The obtained supervisor cannot provide an Optimal solution for all classes of the PNM. Piroddi *et al.*, [21] developed a new method to computes supervisor by combining the siphon techniques and deadlock markings of the PNM for FMSs. A maximally permissive supervisor is obtained in the developed method. However, the computational complexity increases exponentially with the size of the PNM. An iterative mixed integer programming based deadlock detection technique is developed in [22] to enumerate minimal siphons without complete siphons enumerations. The method provides the liveness of the PNM when all the control places are added but cannot ensure the optimal solution of the controlled PNM. In all the methods review [1]–[4], [15]–[17], [19]–[22] have used the control places in the supervisor, which characterized with high restriction on reachable states of the PNM. This implies that too much use of control places in the supervisor may lose concurrent systems properties for FMSs. Our proposed method aims to relax the Petri net model's restriction of reachable states by construction a supervisor using control places and control transitions to maintain the concurrent operation by the processes in the FMSs.

To maintain all the reachable states generated by the uncontrolled PNM in the controlled nets, a supervisor using control transition is introduced in the works of [23]. The methods compute the deadlock markings such that a transition controller is designed for each deadlock marking of the PNM. The challenge faced by the method is where the deadlock markings are returned in the live zone and the nature of the reachability graph of the control nets. The works of [24] modified the method in [23] to reduce the structural complexity of the supervisor for FMSs using transition controller. Deadlocks states are categorized into groups based on their common mark places, such that for each group, a single transition controller needs to control the deadlocks marking in that group. In that case, not all deadlock states are controllable after adding the control transitions, necessitating a recomputation of the reachability graph for the second time to determine livelock markings in the partial control PNM. This iterative process is continued till all the deadlock and livelock markings are controllable. The methods suffer from computational complexities due to several computations of reachable markings of the PNM. Chao *et al.*, [25] developed a supervisor of the PNM for FMS in combined control places and control transitions. Siphon analysis is used to design control places, while a label Petri net is used to implement the control transitions. Simultaneously, the transition controllers are added to ensure that all the uncontrolled PNM states are reached in the controlled PNM. Its limitation is that the reachability of the controlled PNM remains unknown.

Zhang and Uzam developed a new method in [26] that enforce liveness on the PNM using control transitions. A reachability graph is used to computes the deadlocks markings of the PNM for FMSs. Control transitions that are equal to the number of markings in the live zone are designed for each deadlock marking since there is uncertainty on where the deadlock markings will return in the live zone. In this case, too many control transitions are computed in the supervisor, which increases computational and structural complexities. The methods employed a set of covering problems (SCP) to minimize the number of control transitions. The SCP reduces the structural complexity, yet the method has suffered computational complexities. The works of [27] developed a supervisor for FMSs using structural analysis, which utilized some important properties of the PNM to minimize the computational complexity of the supervisor. The method is an iterative procedure that computes first-met uncontrolled transitions, second-met uncontrolled transitions, and n-met uncontrolled transitions. The first-met controller creates Self-loops control arcs to all the markings generated by the first-met uncontrolled transitions. Similarly, the same procedure is repeated for the second-met, n-met uncontrolled transitions. These self-loop control arcs created by the controllers causes works repetitions by the processes that may be possible, which delay the complete processing of the parts. In the proposed paper, the controller creates new markings equal to the number of concurrent states possessed by the FMS, such that the deadlock markings are first projected to the new markings.

The studies of Huang *et al.*, in [28] design a supervisor using control transitions. Deadlocks markings are computed iteratively using a reachability graph by varying the markings of the idle places of the PNM. The deadlock markings are computed iteratively from the lowest possible combination of markings in the idle places. At each iteration step, control transitions are designed for each deadlock markings if it exists. The method greatly simplified the supervisory structure of the PNM. Row and Pan developed a supervisor using control transitions [29] to modify the methods developed in [28] based on reachability graph analysis. The methods minimize the complexity of the supervisor but did not obtain the minimal supervisory structure. The literature in [29] further simplifies the supervisor of FMSs in Row and Pan to obtained the minimal supervisory structure. The methods introduce the iterative computations of crucial deadlock markings by varying the initial markings of the idle places of the PNM for FMSs. Generally, the supervisors computed for these methods developed in [28]–[30] return the deadlock markings to the initial marking.

A new method to computes supervisory structure for FMSs using transition controller is developed by Chen *et al.*, in [31]. The method uses a reachability graph analysis to computes the deadlocks markings in the PNM. It then employed the vector covering approach to computes the minimal number of legal markings by solving integer linear programming problems (ILPPs). The methods return the deadlock markings to the minimal number of legal markings in the reachability graph of the PNM for FMSs. Since the minimal number of legal markings for the large PNM for FMSs are many. It implies that it is difficult to track where the systems may return when entering deadlock states. Moreover, the method is computationally complex, but it guaranteed a minimal supervisory structure. Dong *et al.*, [32] further simplified the method developed in [31] to reduce the computational complexity by avoiding the computation of ILLP. Instead, it employed a vector intersection approach of the deadlock markings to regroup them. A single transition controller is designed to return all the deadlock markings to the minimal legal markings in the reachability graph for each group. The uncertainty that dominates the works for supervisor using control transitions [23]–[33] are what markings in the live zone should the deadlock markings returns, such that it can maintain a smooth operation of the systems.

This paper proposed a new method to design supervisor for FMSs using a Place-Transition Controller (PTC) and a Transition-Place Controller (TPC). A loop marking is computed for each concurrent processes in the FMSs to specify where the deadlock markings return in the live marking without affecting the proper operation of the system. TPC constitutes the input control transitions that are enabled at the deadlock markings and generates new markings in accordance with the number of concurrent processes in the PNM, such that all deadlock markings are first projected to the new markings. Then, the new markings are further projected to the loop markings in the PNM by the action of PTC. This method's main advantage is that the controllers have zero restriction in executing the concurrent works by the PNM for FMSs. Furthermore, the number of controller states is minimal compared with the related method developed in the literature. Finally, our proposed method is efficient and has moderate structural complexity than the related methods in the literature.

The paper is organized as follows. Section II presents the preliminaries of the PNM. Section III describes the dump and loop states. Section IV presents the computation of the Place-Transition/Transition-Place Controller. Section V provides a control policy of the proposed method. While in Section VI, and Section VII presents experimental examples and discussions, respectively. Finally, Section VIII concludes this paper.

II. PRELIMINARIES

A. PETRI NETS

A Petri net is a four-tuple $N = (P, T, F, W)$, where P is the set of place, and T is a transition set in the PNM. Places in the PNM are partitions into two sets as $P = P_A \cup \Theta$, where P_R represents the set of activity places, and Θ represents the shared places. $F \subseteq (P \times T) \cup (T \times P)$ are control arcs connected from places to transitions or transitions to places. $W : (P \times T) \cup (T \times P) \rightarrow \mathbb{N}$ is a weight assigned to the control arcs connected from places to transitions or vice versa. A token is a black dot that is put in places to indicate the marking (state) of the system. If tokens are introduced in $N = (P, T, F, W)$, the PNM is represented as (N, M_0) symbolizing its dynamic structure at the initial marking. Let $y \in P \cup T$ be a node in $N = (P, T, F, W)$. The preset (input) of y is defined as ${}^{\bullet}y = \{l \in P \cup T \mid (l, y) \in F\}$ and its output (postset) of y is defined as $y^{\bullet} = \{l \in P \cup T \mid (y, l) \in F\}$. Let $[N^+]$ and $[N^-]$ be the input and output incidence matrix of N, and defined as $[N^+](p, t) = W(t, p)$ and $[N^-](p, t) = W(p, t)$, respectively. The incidence matrix of the PNM is $[N] = [N^+](p, t) - [N^-](p, t)$.

A marking M of a PNM is a mapping vector $M : P \rightarrow \mathbb{N}$. $M(p)$ represents the number of tokens in place p. With no ambiguity, a marking is some time refer to as state in this paper. $\forall p \in P$ is said to be marked if $M(p) \geq 0$. Marking is represented in a formal sum notation as $\sum_{P_i \in P} M(p_i)p_i$. For example, let a PNM has set of place $P = \{p|p : p_1, \cdots, p_6\}$. If each place p_1 and p_3 have one token and place p_6 has two tokens. Then, the marked places are $M(p_1) = 1$, $M(p_3) = 1$, and $M(p_6) = 2$. The marking of the PNM in formal sum notation is $M = p_1 + p_3 + 2p_6$.

Let $t \in T$ be transition in (N, M_0). t is enabled at a marking M if $\forall p \in t^{\bullet}$, $M(p) \geq W(p, t)$, and it is denoted as $M[t\rangle$. The marking M of the system (N, M_0) is change when the transition t is fired, generating a new marking M' represented as $\forall p \in P, M'(p) = M(p) - W(p, t) + W(t, p)$. For example, if (N, M_0) is at marking $M = p_1 + p_2 + p_6$ with $W(p_1, t) = W(p_2, t) = W(t, p_3) = 1$. This translate that p_1, and p_2 are preset of t, and p_3 is postset of t in (N, M_0). Transition t is enabled since $M(p_1) \geq W(p_1, t)$, and $M(p_2) \geq W(p_2, t)$, when it's fired it is new marking is $M' = p_3 + p_6$. t is said to be dead if $\forall p \in^{\bullet} t$, $W(p, t) \geq M(p)$. A PNM (N, M_0) is said to be dead at M_0 if $\nexists t \in T$ can satisfy $M_0[t\rangle$. t is live if $\forall M \in R(N, M_0)$, $\exists M' \in R(N, M)$, such that $M'[t\rangle$. (N, M_0) is live if $\forall t \in T$, t is live at M_0.

Place invariant and transition invariant are two important structural properties, which play vital role in mathematical computation of (N, M_0). A P-vector is a column vector $I : P \rightarrow \mathbb{Z}$, indexed by P, where $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$. A P-vector I is a place invariant if $I \neq \mathbf{0}$ and $I^T[N] = \mathbf{0}^T$. A P-invariant I is called P-semiflow if every element of I is non-negative. $||I||$ is called the support of P-invariant and is defined as $||I|| = \{p|I(p) \neq 0\}$. The support of P-invariant is called minimal P-invariant if it doesn't contains any other invariant. A T-vector is a column vector $D : P \rightarrow \mathbb{Z}$, indexed by T, where $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$. A T-vector D is called transition invariant if $D \neq \mathbf{0}$ and $[N]D = \mathbf{0}$. The support of T-invariant is defined as $(||D|| = \{t|D(t) \neq 0\})$.

B. MOTIVATION

Concurrency is a property of a system to executes multiple works using shared resources. For example, FMS consists of shared resources, and each shared resource can ensure the correctness of its work. If shared resources operate concurrently in the system to works on the parts for the processes, some parts may be retained by the resources longer than desired and causes the blockage of the system or deadlock. It necessitates controlling the FMSs concurrent operation on different parts to deals with incorrect operations of FMS.

Supposed, we consider the PNM shown in Fig. 1a, which has two concurrent processes that can process two parts at a time. Each process has one input and output place, such that parts P1 and P2 can work independently. Machines M1 and M2 need to work on parts P1 and P2 before the product's output. Robot R1 can load and unload parts P1 and P2 from machines M1 to M2 or vice versa. The processes are independent in terms of input and output sources. However, they shared resources to complete their operation on the parts. At the initial marking of the PNM shown in Fig. 1a, the transitions t_1 and t_5 are enable, which allow the system to process parts P1 and P2 concurrently at a time as desire from the modelling perspective. That means machines M1 can work on part P1 and machine M2 can work on part P2 simultaneously, then R1 can unload P1 from M1 or P2 from M2 that depend on which machine can operate first. The next enable transitions are either t_2, t_5 or t_1, t_6 in the PNM. This process would continue to evolve as it is desired till the system is turned off. However, deadlock occurrence can stop the system earlier than the desired time. Suppose control places are used in constructing the supervisory structure, as shown in Fig. 1b, to control the deadlock occurrence. In that case, it restricts the system's concurrent operation.

III. COMPUTATION OF LOOP MARKING

This section presents the basic concept for the computation of the loop marking in the reachability graph for a PNM (N, M_0). The two markings are the key aspect in designing the controller for the PNM of FMSs. The loop marking is the live marking in the original reachability graph before the controller is added to the (N, M_0). Its primary function is to accept all the deadlock markings into the legal marking. A dump marking is an extra state created by the controller to link the deadlock markings to the loop markings via it. The dump state is the next state after the deadlock markings when the control transition is fired. Generally, our proposed method works on S^4PR class of Petri nets, which is more generalized than other classes of Petri nets.

Definition 1: [37] A system of sequential systems with shared resources (S^4PR) is a generalized connected self-loop Petri net $N = (P, T, F, W)$, where

1) $P = P_A \cup \Theta$, is a place partition such that
   - $P_A = \cup_{i=1}^{n} P_{A_i}$ is called the set of activity places, where $P_{A_i} \neq 0$ and $P_{A_i} \cup P_{A_j} = \emptyset$;
   - $\Theta = \{r_1, r_2, \cdots, r_m | m \in \mathbb{N}\}$ is called the set of resource places;
2) $T = \cup_{i=1}^{n} T_i$ is called the set of transitions, where $\forall (i, j) \in \mathbb{N}$, $i \neq j$, $T_i \neq \emptyset$, and $T_i \cap T_j = \emptyset$;
3) $W = W_A \cup W_\Theta$, where $W_A : (P_A \times T) \cup (T \times P_A) \rightarrow \{0, 1\}$ such that $\forall i, j \in \mathbb{N}, j \neq i, (P_{A_j} \times T_i) \cup (T_i \times P_{A_j}) \rightarrow \{0\}$, and $W_e : (\Theta \times T) \cup (T \times \Theta) \rightarrow \mathbb{N}$;
4) $\forall r \in \Theta$, there exists a unique minimal P-invariant $I_r \in \mathbb{N}^{|p|}$, such that $\{r\} = \|I_r\| \cap \Theta$, $P_A \cap \|I_r\| \neq \emptyset$, and $I_r(r) = 1$, where $\mathbb{N}^{\|P\|}$ is a set of P-dimensional non-negative integer vectors;
5) N is strongly connected.

Definition 2: [5] Let r be a resource in (N, M_0) with $N = (P_R \cup P_A, T, F)$ and I_r be the minimal P-semiflow associated with r. Resource $r^s \in P_R$ is said to be shared if $r^s \in ||I||$, and $\exists j \in \{1, 2, \cdots, n\}$, $(||I|| \setminus \{r^s\}) \subseteq P_{Aj}$. Let Θ denotes the set of r^s in (N, M_0), then $\forall p_i \in P$, $\Theta(p) = 1$ if $r^s \in \Theta$, otherwise $\Theta(p) = 0$.

Definition 3: Let $j \in \{1, 2, \cdots, d\}$ be the number of concurrent processes in (N, M_0) with $N = (P_R \cup P_A, T, F)$. H_j denotes the set of activity places in the j-process of the (N, M_0) such that $P_A = \bigcup_{j=1}^{d} H_j$. $\forall p_i \in P$, $H(p_i) = 1$, if $p_i \in H$ and $\forall p_i \in P$, $H(p_i) = 0$, if $p_i \notin H$

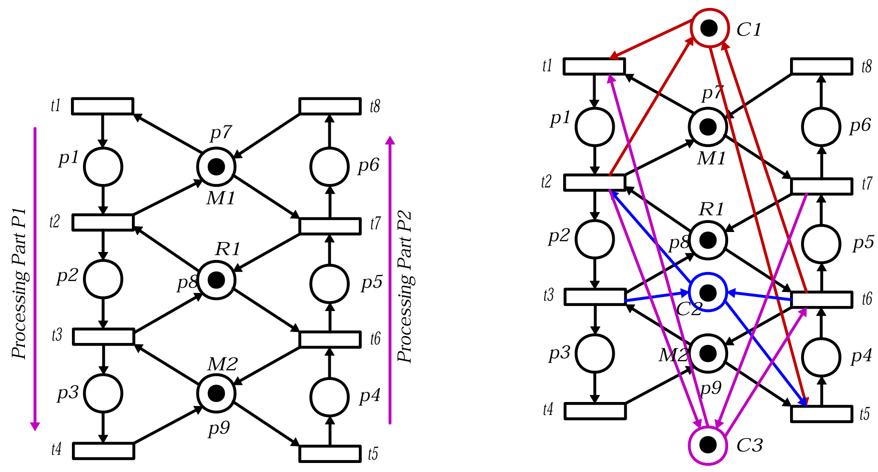
Definition 4: Let $t \in T$ be a transition in (N, M_0). t is called an active transition in (N, M_0) if it enables at the initial marking (i.e., $M_0[t\rangle$). $t \in T$ is said to be sink transition if it consumes token when fired (i.e., it return the system to the previous states in (R, M_0)). Let denotes t^q and t^s be the active and sink transition, respectively.

Definition 5: Let $t^q \in T$ and $t^s \in T$ be the active and sink transition, respectively, and LZ be the set of legal markings in $R(N, M_0)$. A marking M is called a loop marking denoted as M_r^δ if it satisfies $M_r^\delta \in LZ$, $M_r^\delta[t^q\rangle$, and $M_r^\delta[t^s\rangle$. Let Ψ denotes the set of loop markings, then $\Psi = \{M_r^\delta | M_r^\delta \in LZ, t^s, t^q \in T,\ s.t.\ M_r^\delta[t^s\rangle \wedge M_r^\delta[t^q\rangle\}$

Supposed we considered the PNM shown in Fig. 1a, the PNM has two concurrent processes as $j \in \{1, 2\}$. The places in (N, M_0) is categorise into two partition $P = P_A \cup \Theta$, where P_A is further categorised according to their concurrent process i.e., $H_1 = \{p_1, p_2, p_3\}$, and $H_2 = \{p_4, p_5, p_6\}$. The set of resource place is $\Theta = \{p_7, p_8, p_9\}$ that are shared among the two concurrent processes. Three minimal semiflow for the PNM shown in Fig. 1a exists namely as $I_1 : p_1 + p_7 + p_6 \leq 1$, $I_2 : p_2 + p_8 + p_5 \leq 1$, and $I_3 : p_3 + p_9 + p_4 \leq 1$. The support of the minimal semiflow are $||I_1|| = \{p_1, p_6, p_7\}$, $||I_2|| = \{p_2, p_5, p_8\}$, and $||I_3|| = \{p_3, p_4, p_9\}$, respectively. The sink and source transitions for H_1 are t_1 and t_4, respectively. Similarly, the sink and source transitions for H_2 are t_5 and t_8, respectively. Several loop markings may exist due to the sink transition in H_j. Table 1 provides the detail markings of $R(N, M_0)$ for the PNM shown in Fig. 1a, and the loop markings at each process.

A structural analysis is utilized to reduce the loop markings' identification complexity that satisfied Definition 3 in $R(N, M_0)$. Since, the number of loop markings in each H_j increases with the PNM's size. An optimization problem is formulated to minimize computational identification of the loop marking at each concurrent process. The mark activity places selection goal in the loop marking due to each process is to satisfy the constraint in Equ. 1.

$$\sum_{i \in \mathbb{N}} \alpha_i \cdot (I_k(p_i) - H(p_i)) \leq -\beta_k \tag{1}$$



(a)Uncontrolled PNM (b)Controlled PNM

FIGURE 1. PNM control using control places.

TABLE 1. Detail markings of the PNM shown in Fig. 1a.

SN	Details Markings	LZ	DZ	H_1-Loop markings	H_2-Loop markings
1	$p_7+p_8+p_9$	√	-	-	
2	$p_1+p_8+p_9$	√	-	-	
3	$p_2+p_7+p_9$	√	-	-	
4	$p_1+p_2+p_9$	√	-	-	
5	$p_1+p_3+p_8$	√	-	-	
6	$p_2+p_3+p_7$	√	-	√	
7	$p_1+p_2+p_3$	√	-	-	
8	$p_1+p_2+p_4$	-	√	√	-
9	$p_3+p_7+p_8$	√	-	-	
10	$p_2+p_4+p_7$	-	√	-	-
11	$p_1+p_4+p_8$	-	√	-	-
12	$p_1+p_5+p_9$	-	√	-	-
13	$p_1+p_4+p_5$	-	√	-	-
14	$p_4+p_7+p_8$	√	-	-	
15	$p_5+p_7+p_9$	√	-	-	
16	$p_4+p_5+p_7$	√	-	-	
17	$p_4+p_6+p_7$	√	-	-	-
18	$p_5+p_6+p_9$	√	-	-	√
19	$p_4+p_5+p_6$	√	-	-	-
20	$p_6+p_8+p_9$	√	-	-	√

where α_i is a characteristic integer constant-coefficient for the marking of place p_i, $k = \{1, 2, \cdots\}$ represent the number of places invariant, and β_k is a positive constant that can be determine using Equ. 3. The mark resource places selection goal in the loop marking is to satisfy Equ. 2.

$$\sum_{i\in\mathbb{N}} \alpha_i \cdot (I_k(p_i) - \Theta(p_i)) \leq -\beta_r \tag{2}$$

where $\Theta(p_i)$ are marked resource places at the initial marking of (N, M_0), and β_r is a positive constant that can be determine using Equ. 4. Let G_r be the set of resource places in Equs. 1 and 2, and G_A be the set of activity places in Equs. 1 and 2. Then, β_k, and β_r are determined in Equs. 3, and 4, respectively:

$$\beta_k = \begin{cases} |G_{r_i}| - |G_{A_i}|, & r^s \in I_k; \\ |G_{r_i}| - |G_{A_i}| - 1, & r^u \in I_k. \end{cases} \tag{3}$$

where $R = r^s \cup r^u$, and r^u is the set of non-shared resource place in (N, M_0).

$$\beta_r = |G_{r_i}| \tag{4}$$

Based on the Equs. 1, 2, 3, and 4, we can formulate an ILPP to determine the minimum number of mark activity and resource places present in loop marking, denoted as the Minimal Number of Loop Marking Problems (MNLMP).

$$\begin{aligned} & Min \sum_{i\in|P|} \alpha_i \\ & \text{subject to} \sum_{i\in\mathbb{N}} \alpha_i \cdot (I_k(p_i) - H(p_i)) \leq -\beta_k \\ & \qquad \sum_{i\in\mathbb{N}} \alpha_i \cdot (I_k(p_i) - \Theta(p_i)) \leq -\beta_r \\ & \qquad \alpha_i \in \{1, 2, \cdots, \} \\ & \qquad \Theta(p_i), \quad H(p_i) \in P \end{aligned} \tag{5}$$

If $\alpha(p_i) \neq 0$, it means place p_i is marked in $M^{\delta}_{r_i}$. Otherwise, place p_i is unmarked in $M^{\delta}_{r_i}$. The solution to the MNLMP represents the marked activity and resource places in the loop marking. Thanks to the minimization problem that reduces the complexity of searching the vast number of loop markings in $R(N, M_0)$.

IV. COMPUTATION OF PLACE TRANSITION/TRANSITION PLACE CONTROLLER

The Place-Transition and Transition-Place controller write as PTC and TPC, respectively. Both PTC and TPC combine as a single controller connected to a particular concurrent process. At each concurrent process of (N, M_0), the PTC has a single control place and control transition. In contrast, TPC

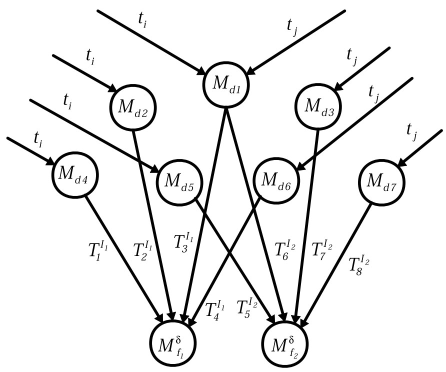


FIGURE 2. **Illustration of dump marking.**

has multiple control transitions that depend on the number of deadlock markings.

Definition 6: $T_i^{I_j}$ is called an additional input control transition if $M_d[T_i^{I_j}\rangle$, and $T_i^{O_j}$ is called an additional output control transition if $\exists p \in M^\delta_{r_j}$, $M^\delta_{r_j}(p) \in (T_i^{O_j})^\bullet$, $\forall j \in \{1, 2, \cdots, d\}$.

Definition 7: Let $T_i^{I_j}$ and $T_i^{O_j}$ be the additional input control transition and output control transition of the Petri net controller for FMSs, respectively. V_j is called the additional control place to the (N, M_0) if $T_i^{I_j}$ and $T_i^{O_j}$ serve as preset and postset to the V_j, respectively.

Definition 8: Let M_d be a deadlock marking of a PNM (N, M_0) with $N = (P, T, F, W)$. M^δ_{fj} is called a dump marking generated as a result of firing $T_i^{I_j}$ from M_d, i.e., $M_d[T_i^{I_j}\rangle M^\delta_{fj}$.

Definition 9: Let $T_i^{I_j}$ be the input control transition and V_j be the control place of the PNM. The TPC is a four-tuple define as $TPC = (\{M^\delta_{fj} \cup V_j\}, T_i^{I_j}, F, W)$, where M^δ_{fj} is the marked places in the dump marking, and V_j is the additional control place. $T_i^{I_j}$ is the additional controlled transitions in the net. $F \subseteq (M^\delta_{fj} \times T_i^{I_j}) \cup (T_i^{I_j} \times V_j)$ is a controller input flow relation represented by arcs with arrow from places to transitions or transitions to places. $W : (M^\delta_{fj} \times T_i^{I_j}) \cup (T_i^{I_j} \times V_j) \rightarrow \mathbb{N}$ is a weight assign to control arc. $j \in \{1, 2, \cdots, d\}$, where d is finite number that represent the concurrent processes in (N, M_0), and $i \in \{1, 2, \cdots\}$.

Definition 10: Let $T_i^{I_j}$ be the input control transition and V_j be the control place of the PNM. The PTC is a four-tuple define as $PTC = (\{M^\delta_{r_j} \cup V_j\}, T_i^{O_j}, F, W)$, where $M^\delta_{r_j}$ is the marked places in loop marking, and V_j is the additional control place. $T_i^{O_j}$ is the additional output controlled transitions in the net. $F \subseteq (M^\delta_{r_j} \times T_i^{O_j}) \cup (T_i^{O_j} \times V_j)$ is a controller input flow relation represented by arcs with arrow from places to transitions or transitions to places. $W : (M^\delta_{r_j} \times T_i^{O_j}) \cup (T_i^{O_j} \times V_j) \rightarrow \mathbb{N}$ is a weight assign to control arc. $j \in \{1, 2, \cdots, n\}$, where n is finite number that represent the concurrent processes in (N, M_0), and $i \in \{1, 2, \cdots\}$.

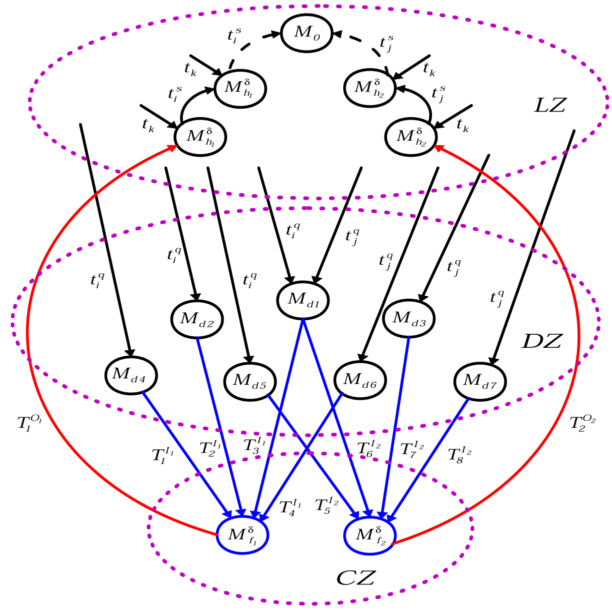


FIGURE 3. **Actions of PTC and TPC on (N, M_0).**

Let considered the markings in the reachability graph as shown in Fig. 2. Markings $M_{d_1}, M_{d_2}, \cdots, M_{d_8}$ are deadlock markings of (N, M_0). Before the addition of the controller, if (N, M_0) enter into any deadlock markings M_{d_1} to M_{d_8}, the system can no longer evolve further. Let assume that (N, M_0) has two concurrent processes (i.e., $j = \{1, 2\}$), and translate to have two set of activity places as H_1 and H_2. Activity places are categorized based on the concurrent processes. Marking is referred to the particular process if it contains more marked activity places from that process. Hence, from Fig. 2, the markings M_{d_1}, M_{d_2}, M_{d_4}, and M_{d_6} are characterized to H_1, and markings M_{d_1}, M_{d_3}, M_{d_5} and M_{d_7} are characterized to H_2. When the TPC is connected to (N, M_0), it implies that two extra markings are created as shown in Fig. 2. The extra markings created by the additional input control transition and control place are known as dump markings (i.e., M^δ_{f1} and M^δ_{f2}), and their primary function is to serve as a transit state from deadlock states to live state. Figure 3 provides the detailed action of TPC and PTC when added to the (N, M_0). The reachability graph of the controlled net by the proposed method are categorized into three regions as shown in Fig. 3, namely as LZ, DZ, and controller Zone denote as CZ. The dump markings M^δ_{f1} and M^δ_{f2} are the CZ marking. The aim of the proposed method is to project the deadlock markings to the loop markings $M^\delta_{r_1}$ and $M^\delta_{r_2}$ by the action of PTC via the dump markings M^δ_{f1} and M^δ_{f2}. The supervisory structure combines of TPC and PTC that provide full control of (N, M_0).

Proposition 1: Let CZ be the set of markings created by the controller. The number of markings in CZ depends on the number of concurrent processes of (N, M_0).

Proof: From Equ. 5, the loop markings are determined based on the concurrent process in (N, M_0). Moreover, only one TPC is added to each concurrent process in (N, M_0). Therefore, the extra places created by the action of TPC depend on the number of concurrent processes possessed by the net systems. Hence, each dump marking is projected to the loop marking in the same process.

Theorem 1: Let CZ be the set controller zone states and Ω be the set of deadlock markings in $R(N, M_0)$. The number of states in CZ is always less than the number of deadlock markings in (N, M_0), i.e., $|CZ| \leq |\Omega|$.

Proof: This is obvious from proposition 1.

V. CONTROL POLICY

This section presents the computations of the PTC and TPC for (N, M_0). The TPC design involves the computations of deadlock markings that serve as input to the TPC, while its output control arc goes to the control place V_j. The output control arc of V_j serves as input to the PTC, and the $M^{\delta}_{r_j}$ serves as output places. Three algorithms are developed to implement our proposed method. Algorithm 2 is used to classify the deadlock marking based on their associates' concurrent processes in (N, M_0). Algorithm 1 presents a step-by-step computation of loop markings from (N, M_0), while algorithm 3 presents TPC and PTC computation that can be included in (N, M_0).

Algorithm 1 Computation of Loop Markings

Input: PNM of an FMS structure suffering from deadlock
Output: Set of loop markings Ψ

Identify the sets of activity places H_j, and set of resource places $\Theta(p_i)$
Compute the minimal semi-flow (I_k) of the PNM (N, M_0)

while $j \leq d$ **do**
  for $(k = 1, z, k++)$ **do**
    /* $k \in \{1, 2, \cdots, z\}$ where z is the number of I in (N, M_0) */
    Compute $\sum_{i \in \mathbb{N}} \alpha_i \cdot (I_k(p_i) - H(p_i)) \leq -\beta_k$
    Compute $\sum_{i \in \mathbb{N}} \alpha_i \cdot (I_k(p_i) - \Theta(p_i)) \leq -\beta_r$
    Solve the minimization problem as presented in Equ. 5
    The solution to the MNLMP represent the loop marking $M^{\delta}_{r_j}$
  end for
  $\Psi = \{M^{\delta}_{r_j}\}$
  $j = j + 1$
end while
Output the set of the loop markings (Ψ)

Proposition 2: Let $\alpha_i \in \mathbb{N}^+$ be the solution for the minimization problems computed using algorithm 1. The solution $\alpha_i \in \mathbb{N}^+$ is the marked places in $M^{\delta}_{r_j}$.

Theorem 2: Let Ψ be the set of $M^{\delta}_{r_j}$ computed using algorithm 1. The Ψ is the minimal set of loop markings that shall guarantee all the deadlock markings are reachable to the initial marking via it.

Algorithm 2 Sorting of Deadlock Markings Based on Their Associated Processes

Input: Reachability graph of PNM
Output: Sets of deadlock markings based on their associate processes Φ_j

Identify the activity places $p \in P_A$, and there concurrent processes it belong, i.e., H_j
Compute the reachability graph $R(N, M_0)$ of (N, M_0)
Identified the set of dead markings from $R(N, M_0)$ denoted as Ω
while $j \leq d$ **do**
  //* where d is the number of concurrent processes in (N, M_0) *//
  while $i \leq n$ **do**
    //* where $n = |\Omega|$ *//
    Compute $M_{d_i} \cap H_j = l, \ \forall M_{d_i} \in \Omega$,
    if $|l| > |M_{d_i}|$ **then**
      $M_{d_i} \in H_j$
    else
      $\Omega = \Omega + M_{d_i}$
    end if
    $\Phi_j = \{M_{d_i}\}$
    $i = i + 1$
  end while
  $j = j + 1$
end while
Output the sets of Φ_j

Proof: Supposed that M^{δ}_r is a loop marking in $R(N, M_0)$ satisfies these constraints $\sum_{i \in \mathbb{N}} \alpha_i \cdot (I_k(p_i) - H(p_i)) \leq -\beta_k$ and $\sum_{i \in \mathbb{N}} \alpha_i \cdot (I_k(p_i) - \Theta(p_i)) \leq -\beta_r$. Since it is true that $\beta_k \geq \beta_r$, then we have $\sum_{i \in \mathbb{N}} \alpha_i \cdot (I_k(p_i) - \Theta(p_i)) \leq \sum_{i \in \mathbb{N}} \alpha_i \cdot (I_k(p_i) - H(p_i))$, which can be simplified to $\sum_{i \in \mathbb{N}} \alpha_i \cdot \Theta(p_i) \leq \sum_{i \in \mathbb{N}} \alpha_i \cdot H(p_i)$. Therefore, it shows that the characteristic coefficient of marked resource places is less than the characteristic coefficient of mark places by the concurrent places in M^{δ}_r. This implies that, M^{δ}_r is the minimal solution at each H_j, $j \in \{1, 2, \cdots\}$. Hence, the overall M^{δ}_r is the minimal number in Ψ due to the concurrent processes.

Theorem 3: Algorithm 3 can provide the optimal control of (N^c, M^c_0) if all the TPC and PTC computed are added to the uncontrolled PNM (N, M_0).

Proof: If an optimal solution is obtained by algorithm 1, then there exists at least a loop marking $M^{\delta}_{r_i}$ that can be used to design a PTC from algorithm 3. Since, all the deadlock markings are rerouted to the loop markings via dump markings. Hence, algorithm 3 provides optimal control for (N^c, M^c_0).

Demonstrated Example:

Let consider the PNM shown in Fig. 4 to illustrate our proposed method. The PNM has 9 places and 8 transitions. Places have the set partitions as $P = P_A \cup \Theta$, where $P_A = \{p_1, p_2, \cdots, p_6\}$ and $\Theta = \{p_7, p_8, p_9\}$. Two concurrent processes possessed by the PNM (i.e., $j = \{1, 2\}$) with $H_1 = \{p_1, p_2, p_3\}$ and $H_2 = \{p_4, p_5, p_6\}$. To demonstrate the

Algorithm 3 Computation of TPC and PTC

Input: The uncontrolled Petri nets model (N, M_0)
Output: Control PNM (N^c, M_0^c)

Compute the reachability graph of the Petri nets (N, M_0)
Identify the deadlock markings in the Petri nets
Identify the set of deadlock markings using algorithm 2
Computes the loop markings using algorithm 1
for $(j = 1, d, j++)$ **do**
  while $\Phi_j \leq \emptyset$ **do**
    Compute the TPC using Definition 9
    $|\Phi_j| = |\Phi_j| - 1$
  end while
  Compute the PTC using Definition 10
end for
Add all the TPC to the (N, M_0)
Add the PTC to the (N, M_0)
Output the control PNM (N^c, M_0^c)

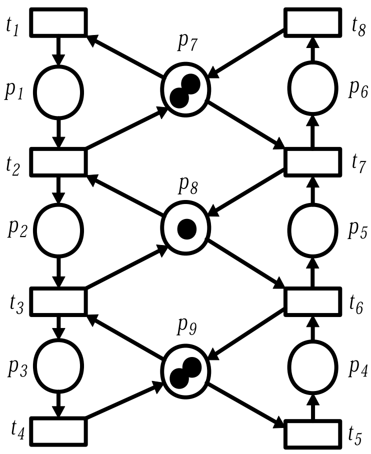


FIGURE 4. A PNM for FMS.

proposed method, we first compute the minimal semi-flow in the PNM shown in Fig. 4. The minimal semi-flow are $I_1 : I_1 + I_6 + I_7 <= 2$; $I_2 : I_2 + I_5 + I_8 <= 1$; and $I_3 : I_3 + I_4 + I_9 <= 2$. Then, at each concurrent processes, we formulate the MNLMP in accordance with Equ. 1 to 5. This implies that, we are to solve two MNLMP for the PNM shown in Fig. 4.

First, for $H_1 = \{p_1, p_2, p_3\}$ with $H_1(p_1) = 1$, $H_1(p_2) = 1$, $H_1(p_3) = 1$, $H_1(p_4) = 0$, $H_1(p_5) = 0$, and $H_1(p_6) = 0$. The set of shared resource places is $\Theta = \{p_7, p_8, p_9\}$ with $\Theta(p_7) = 1$, $\Theta(p_8) = 1$, and $\Theta(p_9) = 1$.

$$\begin{aligned}
&\text{Min } \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 + \alpha_6 + \alpha_7 + \alpha_8 + \alpha_9\\
&\text{Subject to } \alpha_1 + \alpha_6 - \alpha_7 - \alpha_8 - 2\alpha_9 <= -2\\
&\qquad -\alpha_2 - \alpha_3 - \alpha_6 - \alpha_7 <= -1\\
&\qquad \alpha_2 + \alpha_5 - 2\alpha_7 - 2\alpha_9 <= -2\\
&\qquad -\alpha_1 - \alpha_3 - \alpha_5 - \alpha_8 <= -1\\
&\qquad \alpha_3 + \alpha_4 - 2\alpha_7 - \alpha_8 - \alpha_9 <= -2\\
&\qquad -\alpha_1 - \alpha_2 + \alpha_4 + \alpha_9 <= -1\\
&\qquad \alpha_i \in \{1, 2, \cdots\}
\end{aligned}$$

TABLE 2. Details parameters of the TPC for the PNM shown in Fig. 4.

i	$T_i^{I_j}$	$^\bullet(T_i^{I_j})$	$(T_i^{I_j})^\bullet$
1	$T_1^{I_1}$	$2p_1 + p_2 + 2p_4$	V_1
2	$T_2^{I_2}$	$2p_1 + p_5 + 2p_4$	V_2

TABLE 3. Details parameters of the PTC for the PNM shown in Fig. 4.

i	j	$T_i^{O_j}$	$^\bullet(T_i^{O_j})$	$(T_i^{O_j})^\bullet$	V_j	$^\bullet V_j$	$V_j^\bullet$
1	1	$T_1^{O_1}$	V_1	$p_1 + p_2 + p_3 + p_7 + p_9$	V_1	$T_1^{I_1}$	$T_1^{O_1}$
2	2	$T_2^{O_1}$	V_2	$p_4 + p_5 + p_6 + p_7 + p_9$	V_2	$T_2^{I_1}$	$T_2^{O_2}$

The above MNLMP has a solution of $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_7 = \alpha_9 = 1$, and all other parameters are zeros. This implies that the loop marking associated to H_1 is $M_{r_1}^{\delta} = p_1+p_2+p_3+p_7+p_9$. Next, we considered the second concurrent process of the PNM with the set of activity places $H_2 = \{p_4, p_5, p_6\}$ with $H_2(p_1) = 0$, $H_2(p_2) = 0$, $H_2(p_3) = 0$, $H_2(p_4) = 1$, $H_2(p_5) = 1$, and $H_2(p_6) = 1$.

$$\begin{aligned}
&\text{Min } \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 + \alpha_6 + \alpha_7 + \alpha_8 + \alpha_9\\
&\text{Subject to } \alpha_1 + \alpha_6 - \alpha_7 - \alpha_8 - 2\alpha_9 <= -2\\
&\qquad -\alpha_1 - \alpha_4 - \alpha_5 + \alpha_7 <= -1\\
&\qquad \alpha_2 + \alpha_5 - 2\alpha_7 - 2\alpha_9 <= -2\\
&\qquad \alpha_2 - \alpha_4 - \alpha_6 + \alpha_8 <= -1\\
&\qquad \alpha_3 + \alpha_4 - 2\alpha_7 - \alpha_8 - \alpha_9 <= -2\\
&\qquad -\alpha_3 - \alpha_5 - \alpha_6 + \alpha_9 <= -1\\
&\qquad \alpha_i \in \{1, 2, \cdots\}
\end{aligned}$$

The above MNLMP has a solution of $\alpha_4 = \alpha_5 = \alpha_6 = \alpha_7 = \alpha_9 = 1$, and all other parameters are zeros. The loop marking associated with H_2 as $M_{r_2}^{\delta} = p_4+p_5+p_6+p_7+p_9$. The next step is to compute the reachability graph of the PNM and determine the deadlock markings. Two deadlock markings are present in the PNM, as shown in Fig. 4. The set of the deadlock marking is $\Omega = \{M_{d_1}, M_{d_2}\}$ with $M_{d_1} = 2p_1+p_2+2p_4$ and $M_{d_2} = 2p_1+2p_4+p_5$. From algorithm 2, the sets of deadlock markings associated to H_1 and H_2 are $\Phi_1 = \{M_{d_1}\}$, and $\Phi_2 = \{M_{d_2}\}$, respectively. The TPC associated with H_1 has input places from M_{d_1}, and its PTC has output places of $M_{r_1}^{\delta}$. Similarly, the TPC associated with H_2 has input places from M_{d_2}, and its PTC has output places of $M_{r_2}^{\delta}$. Fig. 5 provides the controllers for the two concurrent processes $j = 1$, and $j = 2$, respectively. When the two controllers are included in the PNM shown in Fig. 4, the PNM is live with 97 good Markings.

VI. EXPERIMENTAL EXAMPLES

This section presents an experimental example of an FMS model by PNM to explore the proposed method's applicability.

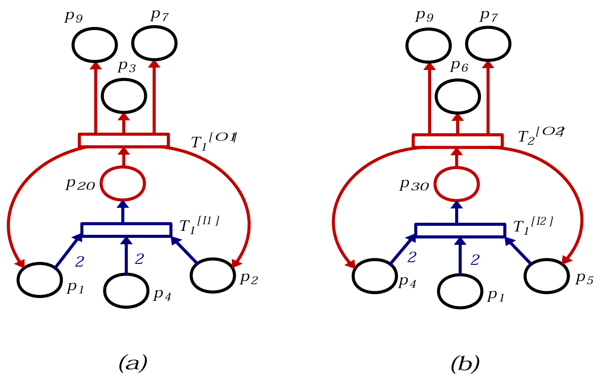


FIGURE 5. (a) Combined TPC and PTC for a PNM shown in Fig. 4 associated with $j = 1$, (b) combined TPC and PTC for a PNM shown in Fig. 4 associated with $j = 2$.

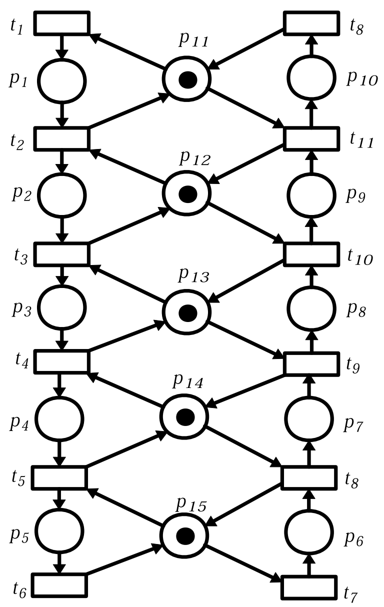


FIGURE 6. A PNM for FMS.

TABLE 4. Details parameters of the TPC for the PNM shown in Fig. 6.

i	$T_i^{I_j}$	$^\bullet(T_i^{I_j})$	$(T_i^{I_j})^\bullet$
1	$T_1^{I_1}$	$p_1+p_2+p_3+p_4+p_6$	V_1
2	$T_2^{I_1}$	$p_1+p_2+p_3+p_6+p_7$	V_1
3	$T_3^{I_2}$	$p_1+p_2+p_6+p_7+p_8$	V_2
4	$T_4^{I_2}$	$p_1+p_6+p_7+p_8+p_9$	V_2

Example 1: Let consider the PNM of FMS shown in Fig. 6. The PNM has 15 places and 12 transitions. The places have the following partitions: $P_A = \{p_1, p_2, \cdots, p_{10}\}$, and $\Theta = \{p_{11}, p_{12}, \cdots, p_{15}\}$. The PNM has 112 reachable markings.

Five minimal semi-flow exist in the PNM: $I_1 = p_1 + p_{10} + p_{11}$, $I_2 = p_2+p_9+p_{12}$, $I_3 = p_3+p_8+p_{13}$, $I_4 = p_4+p_7+p_{14}$, $I_5 = p_5 + p_6 + p_{15}$. The PNM has two concurrent processes,

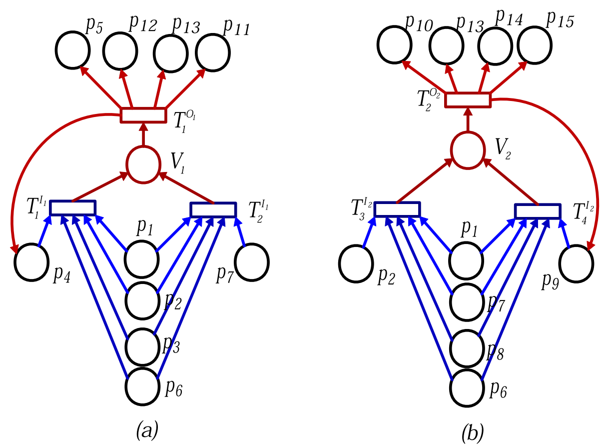


FIGURE 7. (a) Combined TPC and PTC for a PNM shown in Fig. 6 associated with $j = 1$, (b) combined TPC and PTC for a PNM shown in Fig. 6 associated with $j = 2$.

and the sets of their activity places are $H_1 = p_1 + p_2 + p_3 + p_4 + p_5$ and $H_2 = p_6 + p_7 + p_8 + p_9 + p_{10}$. First, we compute the loop markings using algorithm 1. For j_1, in accordance with equations 1 to 5, we formulate this MNLMP:

$$
\begin{aligned}
&\text{Min } \sum\nolimits_{i=1}^{|P|} \alpha_i \\
&\text{Subject to } \alpha_1 + \alpha_{10} - \alpha_{12} - \alpha_{13} - \alpha_{14} - \alpha_{15} <= -2 \\
&\quad -\alpha_2 - \alpha_3 - \alpha_4 - \alpha_5 + \alpha_{10} + \alpha_{11} <= -1 \\
&\quad \alpha_2 + \alpha_9 - \alpha_{11} - \alpha_{13} - \alpha_{14} - \alpha_{15} <= -2 \\
&\quad -\alpha_1 - \alpha_3 - \alpha_4 - \alpha_5 + \alpha_9 + \alpha_{12} <= -1 \\
&\quad \alpha_3 + \alpha_8 - \alpha_{11} - \alpha_{12} - \alpha_{14} - \alpha_{15} <= -2 \\
&\quad -\alpha_1 - \alpha_2 - \alpha_4 - \alpha_5 + \alpha_8 + \alpha_{13} <= -1 \\
&\quad \alpha_4 + \alpha_7 - \alpha_{11} - \alpha_{12} - \alpha_{13} - \alpha_{15} <= -2 \\
&\quad -\alpha_1 - \alpha_2 - \alpha_3 - \alpha_5 + \alpha_7 + \alpha_{14} <= -1 \\
&\quad \alpha_5 + \alpha_6 - \alpha_{11} - \alpha_{12} - \alpha_{13} - \alpha_{14} <= -2 \\
&\quad -\alpha_1 - \alpha_2 - \alpha_3 - \alpha_4 + x_6 + \alpha_{15} <= -1 \\
&\quad \alpha_i \in \{1, 2, \cdots\}
\end{aligned}
$$

The above MNLMP has the solutions of $\alpha_4 = \alpha_5 = \alpha_{11} = \alpha_{12} = \alpha_{13} = 1$, and all other parameters have the value of 0. Similarly, After applying the same procedure on $j = 2$, the loop marking associated with the second process is $M_{r_2}^{\delta} = p_9 + p_{10} + p_{13} + p_{14} + p_{15}$. Then, we compute the reachability graph of the PNM to determine the deadlock markings. The deadlock markings in the net are: $M_{d_1} = p_1 + p_2 + p_3 + p_4 + p_6$, $M_{d_2} = p_1 + p_2 + p_3 + p_6 + p_7$, $M_{d_3} = p_1+p_2+p_6+p_7+p_8$, $M_{d_4} = p_1+p_6+p_7+p_8+p_9$. According to algorithm 2, the set of deadlock markings are $\Psi_1 = \{M_{d_1}, M_{d_1}\}$ and $\Psi_2 = \{M_{d_3}, M_{d_4}\}$. For $j = 1$, the TPC has two input control transitions as $T_1^{I_1}, T_2^{I_1}$. The PTC has one output control transition $T_1^{O_1}$ and one control place V_1.

Similarly, the same procedure is applied for $j = 2$. The TPC has two input transitions as $T_3^{I_2}$, $T_4^{I_2}$, and the PTC has one output control transition $T_2^{O_2}$ and one control place V_2.

TABLE 5. Details parameters of the PTC for the PNM shown in Fig. 6.

i	j	$T_i^{O_j}$	$^\bullet(T_i^{O_j})$	$(T_i^{O_j})^\bullet$	V_j	$^\bullet V_j$	$V_j^\bullet$
1	1	$T_1^{O_1}$	V_1	$p_4+p_5+p_{11}+p_{12}+p_{13}$	V_1	$T_1^{I_1}, T_2^{I_1}$	$T_1^{O_1}$
2	2	$T_2^{O_2}$	V_2	$p_9+p_{10}+p_{13}+p_{14}+p_{15}$	V_2	$T_3^{I_2}, T_4^{I_2}$	$T_2^{O_2}$

TABLE 6. Supervisory structure performance comparison for the PNM shown in Fig. 6.

Methods	No. of control places	No. of control transitions	Total No. of control node	No. of control arcs	No. of reachable markings	No. of controller markings	No. of constraints (N_{LP})	No. of variables (N_r)	System resource utilization (%)
[34]	1	-	1	18	63	-	-	-	56.25
[25]	6	2	8	45	112	-	-	-	100
[18]	5	4	9	46	81	18	34	32	72.32
Proposed method	2	6	8	40	112	2	20	15	100

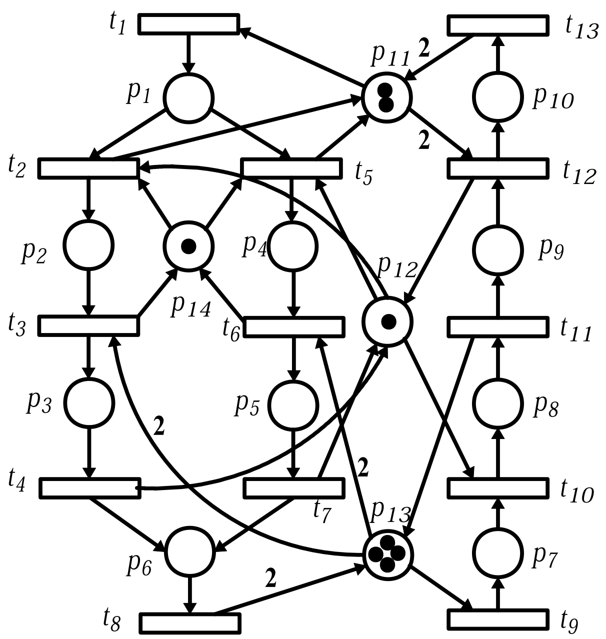


FIGURE 8. A PNM for FMS from [18].

TABLE 7. Details parameters of the TPC for the PNM shown in Fig. 8.

i	$T_i^{I_j}$	$^\bullet(T_i^{I_j})$	$(T_i^{I_j})^\bullet$
1	$T_1^{I_1}$	$2p_1+p_2$	V_1
2	$T_2^{I_1}$	$2p_1+p_4$	V_1
3	$T_3^{I_2}$	$2p_1+3p_7+p_9$	V_2

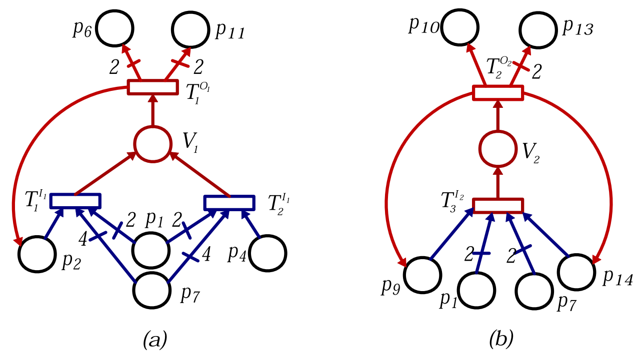


FIGURE 9. (a) Combined TPC and PTC for a PNM shown in Fig. 8 associated with $j = 1$, (b) combined TPC and PTC for a PNM shown in Fig. 8 associated with $j = 2$.

The detailed parameters of the TPC and PTC are presented in Table 4, and Table 5, respectively.

Example 2: Supposed we consider the PNM of FMS shown in Fig. 8. The PNM has 14 places and 13 transitions with 169 reachable makings in its reachability graph. The places have the following partitions: $P_A = \{p_1, p_2, \cdots, p_{10}\}$, and $\Theta = \{p_{11}, p_{12}, \cdots, p_{14}\}$.

Following the same procedure as in Example 1, we formulate this MNLMP. The solution to the MNLMP gives the marked places in $M_{r_1}^{\delta}$.

$$\text{Min } \sum_{i=1}^{|P|} \alpha_i$$

$$\text{Subject to } \alpha_1+\alpha_{10}-\alpha_{11}-\alpha_{12}-\alpha_{13}-\alpha_{14} <= -2$$

$$-\alpha_2-\alpha_3-\alpha_4-\alpha_5-\alpha_6+\alpha_{10}+\alpha_{11} <= -1$$

$$\alpha_2+\alpha_3+\alpha_4+\alpha_5+\alpha_8+\alpha_9-\alpha_{11} -\alpha_{13}-\alpha_{14} <= 3$$

$$-\alpha_1-\alpha_6+\alpha_8+\alpha_9+\alpha_{12} <= -1$$

$$\alpha_3+\alpha_5+\alpha_6+\alpha_7+\alpha_8-\alpha_{11}-\alpha_{12} -\alpha_{14} <= 2$$

$$-\alpha_1-\alpha_2-\alpha_4+\alpha_7+\alpha_8+\alpha_{13} <= -1$$

$$\alpha_2+\alpha_4-\alpha_{11}-\alpha_{12}-\alpha_{13} <= -1$$

$$-\alpha_1-\alpha_3-\alpha_5-\alpha_6+\alpha_{14} <= -1$$

The above MNLMP has optimal solutions of $\alpha_2 = 1$, $\alpha_6 = \alpha_{11} = 2$, and all other parameters have the value of 0. The loop marking associated to $j = 1$ is $M_{r_1}^{\delta} = p_2 + 2p_6 + 2p_{11}$. Similarly, when the same procedure is applied

TABLE 8. Details parameters of the PTC for the PNM shown in Fig. 8.

i	j	$T_i^{O_j}$	$^\bullet(T_i^{O_j})$	$(T_i^{O_j})^\bullet$	V_j	$^\bullet V_j$	$V_j^\bullet$
1	1	$T_1^{O_1}$	V_1	$p_2+2p_6+2p_{11}$	V_1	$T_1^{I_1}, T_2^{I_1}$	$T_1^{O_1}$
2	2	$T_2^{O_2}$	V_2	$p_7+p_9+p_{10}+p_{13}+p_{14}$	V_2	$T_3^{I_2}$	$T_2^{O_2}$

TABLE 9. Supervisory structure performance comparison for the PNM shown in Fig. 8.

Methods	No. of control places	No. of control transitions	Total No. of control node	No. of control arcs	No. of reachable markings	No. of controller markings	No. of constraints (N_{LP})	No. of variables (N_r)	System resource utilization (%)
[34]	1	-	1	26	113	-	-	-	66.86
[25]	6	3	9	35	155	-	-	-	92.81
[18]	5	4	9	34	167	42	70	52	98.82
Proposed method	2	5	7	30	169	2	16	14	100

on $j = 2$, the loop marking associated with the second process is $M_{r_2}^{\delta} = p_7 + p_9 + p_{10} + p_{13} + p_{14}$. Then, we compute the reachability graph of the PNM to determines the deadlock markings. The net model has the following deadlock markings: $M_{d_1} = 2p_1 + p_2$, $M_{d_2} = 2p_1 + p_4$, $M_{d_3} = 2p_1 + 3p_7 + p_9$. The TPC and PTC controllers are design based on algorithm 3. When the controllers are added to the PNM shown in Fig. 8, the PNM is live with 169 reachable marking with the two markings from the controller side. The detailed parameters for the TPC and PTC of the PNM shown in Fig. 8 are shown in Table 7, and Table 8, respectively.

VII. DISCUSSION

This section provides a discussion of the proposed method based on its structural complexity and computational complexity. First, we compared the obtained supervisor performance with some related methods developed in the current literature. The works of [25], [34] used a siphon technique to build a supervisor for PNM of FMSs. However, the supervisor in [25] combined the control places and control transitions to obtain a complete reachable marking for PNM. Moreover, the works in [18] report a related supervisor that both contain additional control places and transitions. The method used reachability graph analysis to computes its elements of a supervisor. Our proposed method is compared with the results obtained in [25], [34], and [18].

Based on the state-of-the-art methods, we have compared the supervisory structure components for our proposed method with other related methods in the literature based on structural complexity and computational complexity. Table 6 provides a detailed number of control places and a number of control transitions in column 1 and column 2, respectively. Column 3 provide the total number of control node (i.e., number of control places plus the number of control transitions) used by each method. The number of additional control arcs is provided in column 4 of Table 6. Column 1-4 gives the total supervisory structural complexity for each method. Column 5 of Table 6 provides the number of reachable markings representing the supervisory structural performance efficiency. The additional number of markings generated by the controllers of the supervisory structure is represented in column 6, aiming to keep the number of the additional controller markings minimal. Columns 7 and 8 provide the total number of constraints and variables to be solved for each method, which amounts to each method's computational complexity. System resource utilization is defined as the ratio of the number of reachable states in the control PNM to that of the reachable states in the uncontrolled PNM. The last column provides system resource utilization for each method. Overall, the method in [34] provides the minimal structural complexity as view in columns 4-5, but has the least reachable states and system resource utilization. Our proposed method has moderate structural complexity as a view from columns 4-5 with a total control node of 7 and 30 control arcs. Our proposed method has 100 percent system resource utilization and less computational complexity. Similarly, the same comparison is applied to Table 9.

Second, we analyzed the computational complexity of the proposed method. The computational complexity of solving MNLMP from algorithm 1 is $\mathcal{O}(\alpha \cdot |P|)$. Since the MNLMP involves solving only one variable α_i for $|P|$-times. In this regard, its computational complexity is polynomial time. For algorithm 2, it's computational complexity is NP-hard since it involves the computation of deadlock markings from the reachability graph. The computational complexity from algorithm 3 is $\mathcal{O}(|V_j|)$-times, which is polynomial, where j is the number of concurrent processes in (N, M_0). In general, the overall computational complexity of the proposed method is polynomial time complexity.

VIII. CONCLUSION

In this paper, we proposed a supervisory structure constructed using control places and control transitions to control the operation of FMSs. Deadlock markings are returned to the live markings via the extra states created by the TPC, while PTC determines the loop markings in the legal markings by solving an integer linear programming problem known as MNLMP. Three algorithms are proposed in this study. The first algorithm computes the minimal number of loop markings from the legal markings in the reachability graph using structural analysis. The second algorithm is used to sort the deadlock markings based on the concurrent processes it associates. The third algorithm is used to construct the TPC and PTC as a supervisory structure responsible for controlling the operation of the FMSs. The final supervisory structure has a zero restriction on the states generated by the controlled systems. This implies that the processes of FMS observe a full concurrent operation as it is desired. The proposed method is efficient as it provides maximally permissive behavior of the PNM. The limitation of the proposed method is that the overall supervisory structure is not minimal, even though it is better than the related works in the literature. Our future works would focus on getting the minimal supervisory structure for the PNM of FMSs using combined PTC and TPC.

REFERENCES

[1] T. Murata and N. Komoda, "Liveness analysis of sequence control specifications described in capacity designated Petri net using reduction," in *Proc. IEEE Int. Conf. Robot. Automat.*, Raleigh, NC, USA, Mar. 1987, pp. 1960–1965, doi: 10.1109/ROBOT.1987.1087786.

[2] G. Alpan and B. Gaujal, "Supervisory control of Petri nets using routing functions: Starvation avoidance issues," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 30, no. 5, pp. 684–695, Oct. 2000, doi: 10.1109/3477.875445.

[3] N. Du, H. Hu, and M. Zhou, "Robust deadlock avoidance and control of automated manufacturing systems with assembly operations using Petri nets," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 4, pp. 1961–1975, Oct. 2020, doi: 10.1109/TASE.2020.2983672.

[4] A. Giua and C. Seatzu, "Modeling and supervisory control of railway networks using Petri nets," *IEEE Trans. Autom. Sci. Eng.*, vol. 5, no. 3, pp. 431–445, Jul. 2008, doi: 10.1109/TASE.2008.916925.

[5] M. Bashir, Z. Li, M. Uzam, A. Al-Ahmari, N. Wu, D. Liu, and T. Qu, "A minimal supervisory structure to optimally enforce liveness on Petri net models for flexible manufacturing systems," *IEEE Access*, vol. 5, pp. 15731–15749, 2017.

[6] M. Bashir, B. B. Muhammad, and Z. Li, "Minimal supervisory structure for flexible manufacturing systems using Petri nets," in *Proc. 2nd Int. Conf. Control, Automat. Robot. (ICCAR)*, Hong Kong, Apr. 2016, pp. 291–296.

[7] M. Bashir, Z. Li, M. Uzam, N. Wu, and A. Al-Ahmari, "On structural reduction of liveness-enforcing Petri net supervisors for flexible manufacturing systems: An algebraic approach," *IMA J. Math. Control Inf.*, vol. 35, no. 4, pp. 1217–1249, Dec. 2018.

[8] M. Bashir and L. Hong, "Global supervisory structure for decentralized systems of flexible manufacturing systems using Petri nets," *Processes*, vol. 7, no. 9, pp. 1–21, 2019.

[9] M. Bashir, "Optimal enforcement of liveness for decentralized systems of flexible manufacturing systems using Petri nets," *Trans. Inst. Meas. Control*, vol. 42, no. 12, pp. 2206–2220, Aug. 2020.

[10] R. Lin, Z. Yu, X. Shi, L. Dong, and E. A. Nasr, "On multi-step look-ahead deadlock prediction for automated manufacturing systems based on Petri nets," *IEEE Access*, vol. 8, pp. 170421–170432, 2020, doi: 10.1109/ACCESS.2020.3022643.

[11] J. Zhou, J. Luo, D. Lefebvre, and Z. Li, "Modeling and scheduling methods for batch production systems based on Petri nets and heuristic search," *IEEE Access*, vol. 8, pp. 163458–163471, 2020, doi: 10.1109/ACCESS.2020.3020805.

[12] H. Kaid, A. Al-Ahmari, E. A. Nasr, A. Al-Shayea, A. K. Kamrani, M. A. Noman, and H. A. Mahmoud, "Petri net model based on neural network for deadlock control and fault detection and treatment in automated manufacturing systems," *IEEE Access*, vol. 8, pp. 103219–103235, 2020, doi: 10.1109/ACCESS.2020.2999054.

[13] A. Al-Ahmari, H. Kaid, Z. Li, and R. Davidrajuh, "Strict minimal siphon-based colored Petri net supervisor synthesis for automated manufacturing systems with unreliable resources," *IEEE Access*, vol. 8, pp. 22411–22424, 2020, doi: 10.1109/ACCESS.2020.2968469.

[14] H. Liu, W. Wu, and H. Yang, "Strong controllable siphon basis-based robust deadlock control for manufacturing systems with multiple unreliable resources," *IEEE Access*, vol. 8, pp. 269–277, 2020, doi: 10.1109/ACCESS.2019.2958331.

[15] R. S. Sreenivas, "On supervisory policies that enforce liveness in a class of completely controlled Petri nets obtained via refinement," *IEEE Trans. Autom. Control*, vol. 44, no. 1, pp. 173–177, Jan. 1999, doi: 10.1109/9.739118.

[16] I. Demongodin and N. T. Koussoulas, "Differential Petri net models for industrial automation and supervisory control," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 36, no. 4, pp. 543–553, Jul. 2006, doi: 10.1109/TSMCC.2005.848154.

[17] R. Davidrajuh, "Extracting Petri modules from large and legacy Petri net models," *IEEE Access*, vol. 8, pp. 156539–156556, 2020, doi: 10.1109/ACCESS.2020.3020213.

[18] Y. Chen, Z. Li, and M. Zhou, "Optimal supervisory control of flexible manufacturing systems by Petri nets: A set classification approach," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 2, pp. 549–563, Apr. 2014, doi: 10.1109/TASE.2013.2241762.

[19] A. Ghaffari, N. Rezg, and X. Xie, "Design of a live and maximally permissive Petri net controller using the theory of regions," *IEEE Trans. Robot. Autom.*, vol. 19, no. 1, pp. 137–141, Feb. 2003, doi: 10.1109/TRA.2002.807555.

[20] M. Uzam and M. Zhou, "An iterative synthesis approach to Petri net-based deadlock prevention policy for flexible manufacturing systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 37, no. 3, pp. 362–371, May 2007, doi: 10.1109/TSMCA.2007.893484.

[21] L. Piroddi, R. Cordone, and I. Fumagalli, "Combined siphon and marking generation for deadlock prevention in Petri nets," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 39, no. 3, pp. 650–661, May 2009, doi: 10.1109/TSMCA.2009.2013189.

[22] Z. W. Li, H. S. Hu, and A. R. Wang, "Design of liveness-enforcing supervisors for flexible manufacturing systems using Petri nets," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, no. 4, pp. 517–526, Jul. 2007, doi: 10.1109/TSMCC.2007.897333.

[23] Y.-S. Huang, T.-H. Chung, and P.-J. Su, "Synthesis of deadlock prevention policy using Petri nets reachability graph technique," *Asian J. Control*, vol. 12, no. 3, pp. 336–346, Feb. 2010, doi: 10.1002/asjc.188.

[24] Y.-S. Huang, Y.-L. Pan, and P.-J. Su, "Transition-based deadlock detection and recovery policy for FMSs using graph technique," *ACM Trans. Embedded Comput. Syst.*, vol. 12, no. 1, pp. 1–13, Jan. 2013, doi: 10.1145/2406336.2406347.

[25] D. Y. Chao, T.-Y. Chen, J.-T. Chen, and K.-C. Wu, "A best deadlock control for S3PMR to reach all states," *Asian J. Control*, vol. 14, no. 1, pp. 278–283, Jan. 2012, doi: 10.1002/asjc.370.

[26] X. Zhang and M. Uzam, "Transition-based deadlock control policy using reachability graph for flexible manufacturing systems," *Adv. Mech. Eng.*, vol. 8, no. 2, pp. 1–9, Feb. 2016, doi: 10.1177/1687814016631503.

[27] M. Bashir, D. Liu, M. Uzam, N. Q. Wu, A. Al-Ahmari, and Z. Li, "Optimal enforcement of liveness to flexible manufacturing systems modeled with Petri nets via transition-based controllers," *Adv. Mech. Eng.*, vol. 10, no. 1, pp. 1–19, 2018, doi: 10.1177/1687814017750707.

[28] Y. S. Huang, T. C. Row, and W. Wu, "Deadlock prevention technique using additional transitions for Petri nets," *J. Chin. Inst. Eng.*, vol. 41, no. 6, pp. 1–11, 2018, doi: 10.1080/02533839.2018.1498019.

[29] T. C. Row and Y. L. Pan, "Maximally permissive deadlock prevention policies for flexible manufacturing systems using control transition," *Adv. Mech. Eng.*, vol. 10, no. 7, pp. 1–10, 2018, doi: 10.1177/1687814018787406.

[30] T.-C. Row, W.-M. Syu, Y.-L. Pan, and C.-C. Wang, "One novel and optimal deadlock recovery policy for flexible manufacturing systems using iterative control transitions strategy," *Math. Problems Eng.*, vol. 2019, pp. 1–12, Mar. 2019, doi: 10.1155/2019/4847072.

[31] Y. Chen, Z. Li, A. Al-Ahmari, N. Wu, and T. Qu, "Deadlock recovery for flexible manufacturing systems modeled with Petri nets," *Inf. Sci.*, vol. 381, pp. 290–303, Mar. 2017, doi: 10.1016/j.ins.2016.11.011.

[32] Y. Dong, Y. Chen, S. Li, M. A. El-Meligy, and M. Sharaf, "An efficient deadlock recovery policy for flexible manufacturing systems modeled with Petri nets," *IEEE Access*, vol. 7, pp. 11785–11795, 2019, doi: 10.1109/access.2018.2889305.

[33] Y. L. Pan, "One computational innovation transition-based recovery policy for flexible manufacturing systems using Petri nets," *Appl. Sci.*, vol. 10, no. 7, p. 2332, 2020, doi: 10.3390/app10072332.

[34] H. Kaid, A. Al-Ahmari, Z. Li, and R. Davidrajuh, "Single controller-based colored Petri nets for deadlock control in automated manufacturing systems," *Processes*, vol. 8, no. 1, p. 21, Dec. 2019, doi: 10.3390/pr8010021.

[35] E. Abouel Nasr, A. M. El-Tamimi, A. Al-Ahmari, and H. Kaid, "Comparison and evaluation of deadlock prevention methods for different size automated manufacturing systems," *Math. Problems Eng.*, vol. 2015, pp. 1–19, Sep. 2015, doi: 10.1155/2015/537893.

[36] H. Kaid, A. Al-Ahmari, Z. Li, and R. Davidrajuh, "Automatic supervisory controller for deadlock control in reconfigurable manufacturing systems with dynamic changes," *Appl. Sci.*, vol. 10, no. 15, p. 5270, Jul. 2020, doi: 10.3390/app10155270.

[37] Y. Hou, M. Zhao, D. Liu, and L. Hong, "An efficient siphon-based deadlock prevention policy for a class of generalized Petri nets," *Discrete Dyn. Nature Soc.*, vol. 2016, pp. 1–12, Jan. 2016, doi: 10.1155/2016/8219424.

MUHAMMAD BASHIR received the B.Eng. degree in electrical engineering from Bayero University, Kano, Nigeria, in 2010, and the M.Sc. degree in electrical and computer engineering from Meliksah University, Kayseri, Turkey, in 2014, and the Ph.D. degree in control theory and control engineering from the School of Electro-Mechanical Engineering, Xidian University, Xi'an, China, in 2018. He is currently an Associate Professor with the School of Electronics and Information, Xi'an Polytechnic University, Xi'an. His current research interests include supervisory control for discrete event systems, flexible manufacturing systems, Petri net applications, and system modeling.

JIAN ZHOU was born in 1983. He received the Ph.D. degree in control theory and control engineering from the Northwestern Polytechnical University, Xi'an, Shaanxi, China, in 2013. He is currently an Associate Professor with the School of Electronics and Information, Xi'an Polytechnic University, Xi'an. His current research interests include application of robot system modeling and advanced control theory, design and application of UAV flight control systems, embedded system design, and research development.

BASHIR BALA MUHAMMAD received the B.Eng. degree in electrical engineering from Bayero University Kano, Nigeria, in 2010, the master's degree in mechatronics engineering from Universiti Teknologi Malaysia (UTM), in 2015, and the Ph.D. degree from the School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, China, in 2019. He is currently a Lecturer with the Department of Mechatronics Engineering, Faculty of Air Engineering, Air Force Institute of Technology, Kaduna, Nigeria.