Transition Cover-Based Design of Petri Net Controllers for Automated Manufacturing Systems

Article *in* IEEE Transactions on Systems, Man, and Cybernetics: Systems · February 2014

DOI: 10.1109/TSMC.2013.2238923

Huixia Liu
Ludong University

KeYi Xing
Xi'an Jiaotong University

Mengchu Zhou
New Jersey Institute of Technology

Transition Cover-Based Design of Petri Net Controllers for Automated Manufacturing Systems

Huixia Liu, Keyi Xing, *Member, IEEE*, MengChu Zhou, *Fellow, IEEE*, Libin Han, and Feng Wang

***Abstract*—In automated manufacturing systems (AMSs), deadlock problems must be well solved. Many deadlock control policies, which are based on siphons or Resource-Transition Circuits (RTCs) of Petri net models of AMSs, have been proposed. To obtain a live Petri net controller of small size, this paper proposes for the first time the concept of transition covers in Petri net models. A transition cover is a set of Maximal Perfect RTCs (MPCs), and the transition set of its MPCs can cover the set of transitions of all MPCs. By adding a control place with the proper control variable to each MPC in an effective transition cover to make sure that it is not saturated, it is proved that deadlocks can be prevented, whereas the control variables can be obtained by linear integer programming. Since the number of MPCs in an effective transition cover is less than twice that of transition vertices, the obtained controller is of small size. The effectiveness of a transition cover is checked, and ineffective transition covers can be transformed into effective ones. Some examples are used to illustrate the proposed methods and show the advantage over the previous ones.**

***Index Terms*—Automated manufacturing systems (AMSs), deadlock control, discrete event system, linear integer programming (LIP), Petri nets, siphons.**

I. Introduction

THIS PAPER addresses a deadlock control problem in automated manufacturing systems (AMSs). An AMS consists of a set of finite resources such as machines, buffers, and robots. Different types of parts enter the system at discrete times and are concurrently processed. All parts processed in the system compete for these finite resources; thus, problems such as blocking, conflict, and deadlocks may occur. In particular, deadlock states imply a global or local stoppage of the system, which leads to catastrophic results in an AMS [2]–[6], [8]–[16], [18]–[27], [30]–[32]. Therefore, to effectively operate an AMS and to make the best use of its system resources [29], it is necessary to develop an efficient deadlock control policy to guarantee that deadlocks never occur in it.

Manuscript received January 21, 2012; revised June 19, 2012 and August 17, 2012; accepted November 9, 2012. Date of publication March 7, 2013; date of current version January 13, 2014. This work was supported in part by the National Nature Science Foundation of China under Grant 60774083 and Grant 50975224 and National Basic Research Program of China under Grant 2011CB302804. This paper was recommended by Associate Editor M. K. Tiwari.

H. Liu, K. Xing, L. Han, and F. Wang are with the State Key Laboratory for Manufacturing Systems Engineering and the Systems Engineering Institute, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: liu.hui.xia@stu.xjtu.edu.cn; kyxing@sei.xjtu.edu.cn; lbhan@sei.xjtu.edu.cn; fwang@sei.xjtu.edu.cn).

M. Zhou is with the Ministry of Education Key Laboratory of Embedded System and Service Computing, Tongji University, Shanghai 200092, China, and also with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102 USA (e-mail: zhou@njit.edu).

Digital Object Identifier 10.1109/TSMC.2013.2238923

It is well known that Petri nets are a powerful tool for modeling and analyzing discrete event systems, particularly AMSs [10], [17], [30]–[32]. Researchers use Petri nets as a formalism to describe AMSs and develop appropriate deadlock resolution methods [2]–[4], [8]–[16], [18]–[27]. Generally, methods derived from a Petri net formalism for dealing with deadlocks are classified into three categories, namely, deadlock detection and recovery [11], [12], deadlock avoidance [12], [18], [20]–[27], and deadlock prevention [2]–[4], [8], [9], [11], [13]–[16], [19], [26], [27]. The first one uses a monitoring mechanism for detecting the deadlock occurrence and a resolution procedure for appropriately preempting some deadlocked resources. Avoidance methods are online control policies that use feedback information on the current resource allocation status and future process resource requirements, to keep the system away from deadlock states. The last one is usually achieved by establishing a static resource allocation policy such that the system can never enter a deadlock state.

This paper focuses on deadlock prevention methods. A number of such methods characterize deadlocks in terms of siphons or resource-transition circuits (RTCs). Deadlocks can be prevented by avoiding empty siphons or saturated RTCs in Petri nets. Xing *et al.* [26] modeled a production system by using production Petri net (PPN) and defined a set of transitions in deadlocks as a set of transitions that cannot fire any more because they are not enabled by resource places. To guarantee that all the transitions of the PPN are not in deadlocks, they defined a Petri net structure related to siphons that must not be empty. When all the resource capacities are more than one, the proposed policy is maximally permissive because it cannot incur restricted deadlocks. Ezpeleta *et al.* [4] described an AMS using a particular class of Petri nets, which is called systems of simple sequential processes with resources (S^3PRs). It is shown that a marked S^3PR net is live if and only if, for each reachable marking from the initial markings, each minimal siphon has at least one token. To prevent each minimal siphon from being empty, they added a control place and related arcs to it, which guarantees the liveness of the controlled system. Chu and Xie [3] exploited the potential of siphons for the analysis of ordinary Petri nets and proposed a mathematical programming approach and a mixed-integer programming approach for checking general Petri nets and structurally bounded Petri nets, respectively, without the complete siphon enumeration of a plant model. The proposed methods in [3] are applied to Petri net modeling of AMSs and provide deadlock prevention and detection methods. The prevention algorithm is presented in detail in [8], where the AMS is modeled by an S^3PR net. The method is an iterative approach consisting of two main stages.

The first stage, which is known as siphon control, adds, for each unmarked minimal siphon, a control place to the original net with its output arcs to the sink transitions of the minimal siphon. The second stage, which is known as augmented siphon control, is to add a control place to the modified net with its output arcs to the source transitions of the resultant net if the resource places are removed. The second stage assures that there are no new unmarked siphons generated when adding the control places of the first stage. To simplify the structure of the controlled Petri nets, Li and Zhou [13] presented the concept of elementary siphons, whose number is linear with the larger of place and transition counts and, thus, much less than that of all strict minimal siphons (SMSs). They then developed a deadlock prevention algorithm by adding a control place only to each elementary siphon.

Xing *et al.* [27] presented the concept of RTCs and used it to characterize deadlock states in AMSs modeled by S^3PRs. An RTC is a circuit that only contains resource places and transitions in S^3PRs. Wu [21] and Wu and Zhou [22] proposed production process circuits in the resource-oriented Petri nets (ROPNs), which are also a special class of circuits that play an important role for the liveness of ROPNs. They are important structural characterization of deadlock in different Petri net models. For a marked S^3PR without center resources, Xing *et al.* [27] derive an optimal Petri-net-based polynomial complexity deadlock avoidance policy. For a marked S^3PR with center resources, by reducing it and applying the design of an optimal deadlock avoidance policy to the reduced one, a suboptimal deadlock avoidance policy is synthesized, and its computation is of polynomial complexity.

Enlightened by the work in [13] and [27], in this paper, we propose a new deadlock prevention policy for S^3PRs based on RTCs. We introduce, for the first time, the concept of transition covers. A transition cover is a subset of maximal perfect RTCs (MPCs), and its transition set can cover the set of transitions of all MPCs in S^3PRs. The number of MPCs in a transition cover is much less than that of all MPCs and polynomially grows with respect to the size of the original net.

By designing a control place with a proper control variable to each MPC in an effective transition cover, we can design live Petri net controllers for S^3PRs, whereas the control variables can be determined by solving a linear integer programming (LIP) problem.

A transition cover for S^3PRs can be obtained by a proposed algorithm with polynomial complexity. However, such obtained transition covers may not be effective. An example in this paper has shown that such an ineffective transition cover corresponds to a set of elementary siphons, and based on it, a live Petri net controller may not exist. To obtain an effective one, a transformation algorithm is presented by which ineffective ones can be transformed into effective ones.

The rest of this paper is organized as follows: Section II reviews basic definitions and properties of Petri nets, S^3PRs, and RTCs. Section III introduces the definitions of transition covers and deadlock controllers. In Section IV, the concept of an effective transition cover is introduced, and an LIP problem is defined for determining the values of control variables. An algorithm for computing a transition cover and a transformation algorithm are presented. A deadlock prevention method based on transition covers is finally proposed in Section IV. Three examples are used to illustrate the proposed method in Section V. Finally, Section VI concludes this paper.

II. Preliminaries

This section presents Petri nets, S^3PRs, and MPCs. The reader is referred to [1], [4], [7], [17], and [27]–[32] for more details.

A. Basic Definitions of Petri Nets and Graphs

A Petri net is a 3-tuple $N = (P, T, F)$, where P and T are finite, nonempty, and disjoint sets. P is a set of places, and T is a set of transitions. $F \subseteq (P \times T) \cup (T \times P)$ is a set of directed arcs. Given a Petri net $N = (P, T, F)$ and a vertex $x \in P \cup T$, the preset of x is defined as ${}^\bullet x = \{y \in P \cup T | (y, x) \in F\}$, and the postset of x is defined as $x^\bullet = \{y \in P \cup T | (x, y) \in F\}$. The notation can be extended to a set. For example, let $X \subseteq P \cup T$, then ${}^\bullet X = \cup_{x \in X} {}^\bullet x$ and $X^\bullet = \cup_{x \in X} x^\bullet$. A state machine is a Petri net in which each transition has exactly one input place and one output place, i.e., $\forall t \in T$, $|t^\bullet| = |{}^\bullet t| = 1$.

A marking or a state of N is a mapping $M : P \to \mathbb{Z}^+$, where $\mathbb{Z}^+$ is the nonnegative integer set. Given a place $p \in P$ and a marking M, $M(p)$ denotes the number of tokens in p at M, and we use $\Sigma_{p \in P} M(p)p$ to denote vector M. Let $S \subseteq P$ be a set of places, the sum of tokens in all places of S at M is denoted by $M(S)$, i.e., $M(S) = \Sigma_{p \in S} M(p)$. A Petri net N with an initial marking M_0 is called a marked Petri net or a net for simplicity, which is denoted by (N, M_0).

A transition $t \in T$ is enabled at a marking M, which is denoted by $M[t >$, if $\forall p \in {}^\bullet t$, $M(p) > 0$. An enabled transition t at M can be fired, resulting in a new marking M', which is denoted by $M[t > M'$, where $M'(p) = M(p) - 1$, $\forall p \in {}^\bullet t \setminus t^\bullet$; $M'(p) = M(p) + 1$, $\forall p \in t^\bullet \setminus {}^\bullet t$; and otherwise, $M'(p) = M(p)$, $\forall p \in P - \{{}^\bullet t \setminus t^\bullet, t^\bullet \setminus {}^\bullet t\}$. A sequence of transitions $\alpha = t_1 t_2 \ldots t_k$, $t_i \in T$, $i \in \mathbb{N}_k = \{1, 2, \ldots, k\}$, is feasible from a marking M, if there exists $M_i[t_i > M_{i+1}$, $i \in \mathbb{N}_k$, where $M_1 = M$, and M_i is called a reachable marking from M. Let $R(N, M_0)$ denote the set of all reachable markings of N from M_0.

A transition t is live if $\forall M \in R(N, M_0)$, $\exists M' \in R(N, M)$ such that $M'[t >$ holds. It is dead under M if there is no reachable marking from M that enables t. A net is live if every transition is live.

Let $P_1 \subseteq P$ and $T_1 \subseteq T$. The subnet generated by P_1 and T_1, which is denoted by $N[P_1, T_1]$, is a Petri net $N[P_1, T_1] = (P_1, T_1, F_1)$, where $F_1 = F \cap ((P_1 \times T_1) \cup (T_1 \times P_1))$.

The composition of two Petri nets $N_i = (P_i, T_i, F_i)$, $i \in \{1, 2\}$, via the same elements, which is denoted by $N_1 \otimes N_2$, is a Petri net $N_1 \otimes N_2 = (P, T, F)$, where $P = P_1 \cup P_2$, $T = T_1 \cup T_2$, and $F = F_1 \cup F_2$. Two marked Petri nets $(N_i, M_{i0}) = (P_i, T_i, F_i, M_{i0})$, $i \in \{1, 2\}$, are compatible if $\forall p \in P_1 \cap P_2$, $M_{10}(p) = M_{20}(p)$. The composition of two compatible marked Petri nets (N_1, M_{10}) and (N_2, M_{20}) is a marked Petri net $(N_1, M_{10}) \otimes (N_2, M_{20}) = (P, T, F, M_0)$, where $(P, T, F) = N_1 \otimes N_2$; $M_0(p) = M_{i0}(p)$ if $p \in P_i$, $i \in \{1, 2\}$.

A digraph $G(V, A)$ consists of a set V of vertices and a set A of arcs, in which A can be regarded as a set of ordered pairs of vertices, i.e., $A \subseteq V \times V$. We consider only graphs without self-loops and multiple arcs in this paper. We write $(x, y) = e \in A$ and call x the initial vertex and y the terminal vertex of e, respectively. We refer to both x and y as the endpoints of the arc e.

A chain in $G(V, A)$ is a sequence of vertices $\alpha = x_0x_1x_2 \ldots x_q$ such that either $e_k = (x_{k-1}, x_k) \in A$ or $e_k = (x_k, x_{k-1}) \in A$. q is its length. It is closed if its initial and terminal vertices coincide. It is simple (elementary) if it does not contain the same arcs (vertices) twice or more. A cycle is a closed simple chain. A walk is a chain $\alpha = x_0x_1x_2 \ldots x_q$ such that $e_k = (x_{k-1}, x_k)$. A path is a simple walk. A circuit is a closed walk. A circuit of length 2 is called a double edge. Let $c_1 = x_1x_2 \ldots x_n$ and $c_2 = y_1y_2 \ldots y_m$ be two chains and $x_n = y_1$. Then, $c = x_1x_2 \ldots x_ny_2 \ldots y_m$ is, again, a chain, which is called as the union of c_1 and c_2, which is denoted by $c = c_1 \cup c_2$.

If $G(V, A)$ is a digraph, then $G^0(V, A^0)$ denotes the underlying undirected graph that is obtained by ignoring the direction of the arcs and identifying double edges. A directed graph $G(V, A)$ is strongly connected if for all $x, y \in V$, there is a path from x to y and a path from y to x.

Let α be a simple chain in $G(V, A)$. Then, we define the arc-indexed vector C (with coordinates $C(e), e \in A$) by $C(e) = +1$ if $e = e_k = (x_{k-1}, x_k) \in \alpha$, $C(e) = -1$ if $e = e_k = (x_k, x_{k-1}) \in \alpha$, and $C(e) = 0$ if $e \notin \alpha$.

The cycle space ϑ of $G(V, A)$ is the subspace of $\mathbb{R}^{|A|}$ that is generated by the vectors associated with cycles of $G(V, A)$, where $\mathbb{R}$ is the real number set. A circuit basis is a basis of the cycle space ϑ of $G(V, A)$ exclusively consisting of elementary circuits. By [1], a strongly connected digraph $G(V, A)$ has a circuit basis. A minimal circuit basis is a circuit basis with a minimal length.

According to [7], the dimension of the cycle space ϑ is $d = |A| - |V| + c(G^0)$, where $c(G^0)$ denotes the number of connected components of $G^0(V, A^0)$. An algorithm for computing a minimal circuit basis in $G(V, A)$ is outlined in [7], and its time complexity is, at most, $O(d|A|^2|V|)$.

B. S^{3}PRs

S^{3}PRs are first developed in [4] for modeling AMSs with flexible routes and defined in a recursive way.

Definition 1 [4]: A simple sequential process (S^{2}P) is a Petri net $N = (P \cup P_0, T, F)$, where (1.1) $P \neq \emptyset$, $p \in P$ is called an operation place, $P_0 = \{p\}$, $P \cap P_0 = \emptyset$, and p_0 is called a process idle place; (1.2) N is a strongly connected state machine; and (1.3) every circuit of N contains p_0.

Definition 2 [4]: A simple sequential process with resources (S^{2}PR) is a Petri net $N = (P \cup P_0 \cup P_R, T, F)$, where (2.1) the subset generated by $P \cup P_0$ and T is an S^{2}P; (2.2) $P_R \neq \emptyset$ and $(P \cup P_0) \cap P_R = \emptyset$, where $r \in P_R$ is called a resource place; (2.3) $\forall p \in P$, $\forall t \in {}^\bullet p$, and $\forall t' \in p^\bullet$, ${}^\bullet t \cap P_R = t'^\bullet \cap P_R = \{r\}$, which is denoted by $R(p) = r$; (2.4) the following two statements are true: a) $\forall r \in P_R$, ${}^{\bullet\bullet}r \cap P = r^{\bullet\bullet} \cap P \neq \emptyset$ and b) $\forall r \in P_R$, ${}^\bullet r \cap r^\bullet = \emptyset$; and (2.5) ${}^{\bullet\bullet}(p_0) \cap P_R = (p_0)^{\bullet\bullet} \cap P_R = \emptyset$, where $\{p_0\} = P_0$.

The notation $R(p)$ can be extended to a set. For example, let $E \subseteq P$, then $R(E) = \cup_{p \in E} R(p)$. $H(r) = \{p \in P | R(p) = r\}$ is called as the holder set of r.

Definition 3 [4]: A system of S^{2}PR, which is called an S^{3}PR for short, is recursively defined as follows. (3.1) An S^{2}PR is an S^{3}PR. (3.2) Let $N_i = (P_i \cup P_{0i} \cup P_{Ri}, T_i, F_i)$, $i \in \{1, 2\}$, be two S^{3}PRs, such that $(P_1 \cup P_{01}) \cap (P_2 \cup P_{02}) = \emptyset$, $P_{R1} \cap P_{R2} \neq \emptyset$, and $T_1 \cap T_2 = \emptyset$. The net $N = N_1 \otimes N_2 = (P \cup P_0 \cup P_R, T, F)$ resulting from the composition of N_1 and N_2 via $P_{R1} \cap P_{R2}$ is also an S^{3}PR.

Let N be an S^{3}PR. Its acceptable initial marking M_0 must satisfy that 1) $M_0(p_0) \geq 1$, $\forall p_0 \in P_0$; 2) $M_0(p) = 0$, $\forall p \in P$; and 3) $M_0(r) \geq 1$, $\forall r \in P_R$, where $M_0(r)$ is the capacity of resource r.

Let $(N, M_0) = (P \cup P_0 \cup P_R, T, F, M)$ be a marked S^{3}PR and a transition $t \in T$; let ${}^{(o)}t$ and $t^{(o)}$ denote the input and output operations or the process idle places of t, respectively; and let ${}^{(r)}t$ and $t^{(r)}$ denote the input and output resource places of t, respectively. The notation can be extended to a set. For example, let $Y \subset T$, then ${}^{(o)}Y = \cup_{t \in Y} {}^{(o)}t$ and $Y^{(o)} = \cup_{t \in Y} t^{(o)}$. For a given marking $M \in R(N, M_0)$, t is process enabled at M if $M({}^{(o)}t) > 0$, and t is resource enabled at M if $M({}^{(r)}t) > 0$. Only transitions that are process and resource enabled at the same time can be fired. The finiteness of the resource capacities and the initial marking of places in P_0 imply that the set of distinct reachable markings is finite.

Let $N = (P \cup P_0 \cup P_R, T, F)$ be an S^{3}PR and x and y be two nodes in $P \cup T$. If there exists a path in N from x to y with a length greater than 1, which does not contain any place in $P_0 \cup P_R$, we say that x is previous to y in N. This fact is denoted by $x < y$. The fact that x is not previous to y in N is denoted by $x \not< y$. Let $Z \subseteq (P \cup T)$ be a set of nodes of N. Then, we say that x is previous to Z in N if and only if there exists a node $z \in Z$ such that $x < z$, denoted by $x < Z$. The fact that x is not previous to Z in N is denoted by $x \not< Z$.

C. MPCs

An S^{3}PR $N = (P \cup P_0 \cup P_R, T, F)$ is a digraph in which the vertex set consists of the set of places $P \cup P_0 \cup P_R$ and the set of transitions T. Let θ be a circuit in N. θ is called an RTC if it contains only resource places and transitions. Let $\Re[\theta]$ and $\Im[\theta]$ denote the sets of all resource places and all transitions in θ, respectively. It is clear that an RTC θ does not contain any operation places and is uniquely determined by its resource set $\Re[\theta]$ and transition set $\Im[\theta]$. Hence, θ can be denoted by $\theta = \langle \Re[\theta], \Im[\theta] \rangle$. The notations $\Re[\theta]$ and $\Im[\theta]$ can be extended to a set of RTCs. For example, let $\aleph$ be a set of RTCs, then $\Re[\aleph] = \cup_{\theta \in \aleph} \Re[\theta]$ and $\Im[\aleph] = \cup_{\theta \in \aleph} \Im[\theta]$.

Let $\Psi(R)$ denote the set of all RTCs with resource set $R = \Re[\theta]$, and $\theta_1, \theta_2 \in \Psi(R)$. Then, $\theta_1 \cup \theta_2 \in \Psi(R)$ and $\Psi(R)$ contain a unique maximal RTC with resource set R. Any RTC with resource set R is a subcircuit of the maximal RTC with resource set R.

An RTC θ is *perfect* if $({}^{(o)}\Im[\theta])^\bullet = \Im[\theta]$. Let $\Omega(R)$ denote the set of all perfect RTCs (PRTCs) with resource set $R = \Re[\theta]$, and $\theta_1, \theta_2 \in \Omega(R)$. Then, $\theta_1 \cup \theta_2 \in \Omega(R)$. Therefore, $\Omega(R)$ contains a unique MPC, which is denoted by $\delta(R)$. Any PRTC with resource set R is a subcircuit of $\delta(R)$. If $\Omega(R) = \emptyset$, then $\delta(R) = \emptyset$. Let Θ denote the set of all MPCs in an S^{3}PR N throughout this paper.

Let θ be an RTC in N. Then, $\langle \Re[\theta], \Re[\theta]^\bullet \cap {}^\bullet\Re[\theta] \rangle$ is strongly connected and is a maximal RTC with resource set $\Re[\theta]$. For simplicity, let $\gamma(\theta)$ denote $\langle \Re[\theta], \Re[\theta]^\bullet \cap {}^\bullet\Re[\theta] \rangle$. If $\gamma(\theta)$ is perfect, then $\gamma(\theta)$ is the MPC with resource set $\Re[\theta]$, that is, $\delta(\Re[\theta]) = \gamma(\theta)$; if $\gamma(\theta)$ is not perfect, let $V = \{t \in \Im[\gamma(\theta)] | ({}^{(o)}t)^\bullet \not\subset \Im[\gamma(\theta)]\}$. Delete all transitions in V and their related arcs from $\gamma(\theta)$ and obtain θ'. If θ' is strongly connected, then it is the MPC with resource set $\Re[\theta]$, i.e., $\delta(\Re[\theta]) = \theta'$; otherwise, $\delta(\Re[\theta]) = \emptyset$.

An MPC θ is said to be *saturated* under a marking M iff $M({}^{(o)}\Im[\theta]) = M_0(\Re[\theta])$.

Proposition 1 [27]: A marked S^{3}PR (N, M_0) is live if and only if no MPC of N is saturated at any reachable marking of (N, M_0).

Let θ be an MPC and t be a transition in a marked S^{3}PR (N, M_0). t is called an output transition of θ iff its firing decreases tokens in ${}^{(o)}\Im[\theta]$.

RTCs are only related to the transitions and resource places. If a transition is in an RTC, then it must have input and output resource places. Hence, a transition without input or output resource places cannot be in any RTC. Any transition in an RTC is in $P_R^\bullet \cap {}^\bullet P_R$. Hence, to find MPCs, we only need to consider the subnet of N, as defined next.

Definition 4 [28]: Let $N = (P \cup P_0 \cup P_R, T, F)$ be an S^{3}PR. The resource-transition net of N, which is denoted by N_R, is a subnet generated by P_R and $P_R^\bullet \cap {}^\bullet P_R$. That is, $N_R = N[P_R, P_R^\bullet \cap {}^\bullet P_R]$.

III. Transition Cover and Controller

A. Transition Cover

Definition 5: Let N be an S^{3}PR and Θ be the set of MPCs in N, $\Gamma \subseteq \Theta$, and $\theta \in \Theta$. Γ is *a transition cover* of θ or Γ *covers* θ if $\Im[\theta] \subseteq \Im[\Gamma]$. Γ is *minimal* if no proper subset of Γ is a transition cover of θ.

Γ is said to be a *transition cover* of N if $\forall \theta \in \Theta$, $\Im[\theta] \subseteq \Im[\Gamma]$.

Trivially, Θ is a transition cover of N.

Example 1: Consider a marked S^{3}PR (N, M_0) shown in Fig. 1. $\Theta = \{\theta_1, \theta_2, \theta_3\}$, where $\theta_1 = r_1 t_{23} r_2 t_{12} r_1$, $\theta_2 = r_2 t_{22} r_3 t_{13} r_2$, and $\theta_3 = r_1 t_{23} r_2 t_{22} r_3 t_{13} r_2 t_{12} r_1$. Let $\Gamma_1 = \{\theta_1, \theta_2\}$ and $\Gamma_2 = \{\theta_3\}$. Since $\Im[\theta_1] \cup \Im[\theta_2] = \{t_{12}, t_{23}\} \cup \{t_{13}, t_{22}\} = \{t_{12}, t_{13}, t_{22}, t_{23}\} = \Im[\theta_3] = \Im[\Theta]$, Γ_1 and Γ_2 are two transition covers of N and minimal ones of θ_3.

B. Controller Design Based on Transition Cover

In the following discussion, let $(N, M_0) = (P \cup P_0 \cup P_R, T, F, M)$ be a marked S^{3}PR, $\theta \in \Theta$, and $\Gamma \subseteq \Theta$.

Let $I(\theta)$ denote the set of transitions to be previous to ${}^{(o)}\Im[\theta]$ in $P_0^\bullet$; $O(\theta)$ denote the set of output transitions of θ that are not previous to ${}^{(o)}\Im[\theta]$; and $Z(\theta)$ denote the set of transitions that are not previous to ${}^{(o)}\Im[\theta]$, but there exist other transitions having the same input operation places to be

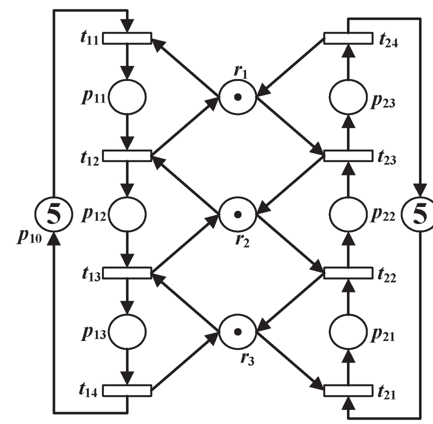


Fig. 1. Marked S^{3}PR (N, M_0).

previous to ${}^{(o)}\Im[\theta]$. That is, $I(\theta) = \{t | t \in P_0^\bullet$ and $t < {}^{(o)}\Im[\theta]\}$, $O(\theta) = \{t | t$ is an output transition of θ, but $t \nless {}^{(o)}\Im[\theta]\}$, and $Z(\theta) = \{t | t \nless {}^{(o)}\Im[\theta]$, and $\exists t_1 \in ({}^{(o)}t)^\bullet$, $t_1 < {}^{(o)}\Im[\theta]\}$.

Definition 6: A *Petri net controller* for θ is defined as follows:

$$(C_\theta, M_\theta) = (\{p_\theta\}, T_\theta, F_\theta, M_\theta)$$

where p_θ is a control place corresponding to θ; and its initial marking is $M_\theta(p_\theta) = M_0(\Re[\theta]) - \xi_\theta$, where $\xi_\theta \in [1, M_0(\Re[\theta]) - 1]$ is an integer, which is called a control variable. $T_\theta = I(\theta) \cup O(\theta) \cup Z(\theta)$, and $F_\theta = \{(p_\theta, t) | t \in I(\theta)\} \cup \{(t, p_\theta) | t \in O(\theta) \cup Z(\theta)\}$.

The determination procedure of control variables for deadlock prevention will be given in the next section.

Definition 7: A *Petri net controller* for (N, M_0) with respect to Γ is defined as follows:

$$(C_\Gamma, M_\Gamma) = \otimes_{\theta \in \Gamma}(C_\theta, M_\theta) = (P_\Gamma, T_\Gamma, F_\Gamma, M_\Gamma)$$

where $P_\Gamma = \{p_\theta | \theta \in \Gamma\}$ is a set of control places, $T_\Gamma = \cup_{\theta \in \Gamma} T_\theta$, $F_\Gamma = \cup_{\theta \in \Gamma} F_\theta$, $M_\Gamma(p_\theta) = M_\theta(p_\theta)$, and $(C_\theta, M_\theta) = (\{p_\theta\}, T_\theta, F_\theta, M_\theta)$ is the Petri net controller for θ in Definition 6.

Let $(CN_\Gamma, M_{\Gamma 0})$ denote the controlled Petri net with respect to Γ, that is, (N, M_0) controlled by (C_Γ, M_Γ). Then

$$\begin{aligned}(CN_\Gamma, M_{\Gamma 0}) &= (N, M_0) \otimes (C_\Gamma, M_\Gamma) \\ &= (P \cup P_0 \cup P_R \cup P_\Gamma, T, F \cup F_\Gamma, M_{\Gamma 0})\end{aligned}$$

where $M_{\Gamma 0}(p) = M_0(p)$, $\forall p \in P \cup P_0 \cup P_R$, and $M_{\Gamma 0}(p) = M_\Gamma(p)$, $\forall p \in P_\Gamma$.

Lemma 1: Let $(CN_\Gamma, M_{\Gamma 0})$ be the controlled Petri net with respect to Γ. Then, $\forall \theta \in \Gamma$, θ is not saturated at any reachable marking of $(CN_\Gamma, M_{\Gamma 0})$.

Proof: Let M be a reachable marking of $(CN_\Gamma, M_{\Gamma 0})$. Then, by Definition 6, we have $M({}^{(o)}\Im[\theta]) \le M_0(\Re[\theta]) - \xi_\theta \le M_0(\Re[\theta]) - 1$. That is, θ is not saturated at M. ■

In $(CN_\Gamma, M_{\Gamma 0})$, let ${}^{(c)}t$ and $t^{(c)}$ denote the input and output control places of t, respectively. Then, ${}^\bullet t = {}^{(o)}t \cup {}^{(r)}t \cup {}^{(c)}t$ and $t^\bullet = t^{(o)} \cup t^{(r)} \cup t^{(c)}$. Let $M \in R(CN_\Gamma, M_{\Gamma 0})$, t is said to be control enabled at M if $\forall p \in {}^{(c)}t$, $M(p) \ge 1$. Only transitions that are process, resource, and control-enabled can fire at M.

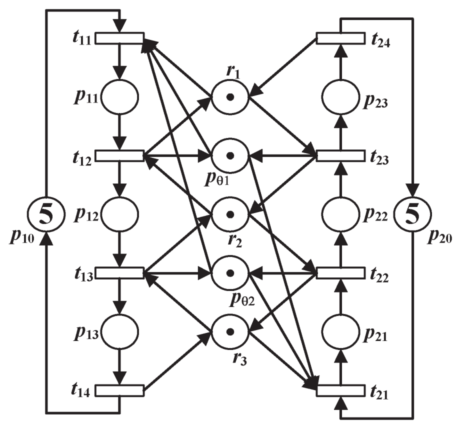


Fig. 2. Controlled Petri net of (N, M_0) shown in Fig. 1.

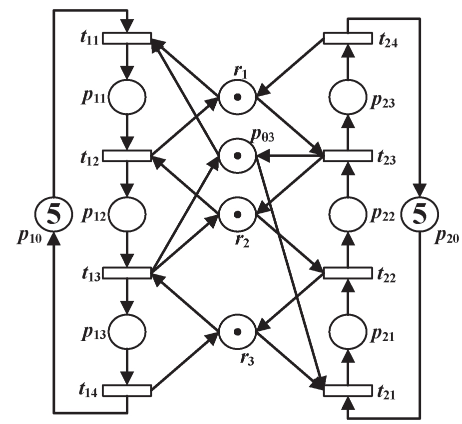


Fig. 3. Controlled Petri net of (N, M_0) shown in Fig. 1.

Let $D(CN_\Gamma, M)$ denote the set of process-enabled but dead transitions at M.

Example 2: Reconsider the marked S^{3}PR (N, M_0) shown in Fig. 1. From Example 1, $\Theta = \{\theta_1, \theta_2, \theta_3\}$. Let $(C_{\theta 1}, M_{\theta 1}) = (\{p_{\theta 1}\}, T_{\theta 1}, F_{\theta 1}, M_{\theta 1})$ and $(C_{\theta 2}, M_{\theta 2}) = (\{p_{\theta 2}\}, T_{\theta 2}, F_{\theta 2}, M_{\theta 2})$ be the Petri net controllers for θ_1 and θ_2, respectively. By Definition 6, $M_{\theta 1}(p_{\theta 1}) = M_0(\Re[\theta_1]) - \xi_{\theta 1} = 2 - \xi_{\theta 1}$. By $1 \le \xi_{\theta 1} \le M_0(\Re[\theta_1]) - 1 = 1$, we have that $\xi_{\theta 1} = 1$ and $M_{\theta 1}(p_{\theta 1}) = 1$. $I(\theta_1) = \{t_{11}, t_{21}\}$, $O(\theta_1) = \{t_{12}, t_{23}\}$, and $Z(\theta_1) = \emptyset$. $T_{\theta 1} = I(\theta_1) \cup O(\theta_1) \cup Z(\theta_1) = \{t_{11}, t_{12}, t_{21}, t_{23}\}$. $F_{\theta 1} = \{(p_{\theta 1}, t_{11}), (p_{\theta 1}, t_{21}), (t_{12}, p_{\theta 1}), (t_{23}, p_{\theta 1})\}$. Similarly, $M_{\theta 2}(p_{\theta 2}) = M_0(\Re[\theta_2]) - \xi_{\theta 2} = 2 - \xi_{\theta 2}$, where $1 \le \xi_{\theta 2} \le 1$, or $\xi_{\theta 2} = 1$. In addition, $M_{\theta 2}(p_{\theta 2}) = 1$, $I(\theta_2) = \{t_{11}, t_{21}\}$, $O(\theta_2) = \{t_{13}, t_{22}\}$, $Z(\theta_2) = \emptyset$, $T_{\theta 2} = I(\theta_2) \cup O(\theta_2) \cup Z(\theta_2) = \{t_{11}, t_{13}, t_{21}, t_{22}\}$, and $F_{\theta 2} = \{(p_{\theta 2}, t_{11}), (p_{\theta 2}, t_{21}), (t_{13}, p_{\theta 2}), (t_{22}, p_{\theta 2})\}$.

Then, the Petri net controller $(C_{\Gamma 1}, M_{\Gamma 1})$ with respect to $\Gamma_1 = \{\theta_1, \theta_2\}$ by Definition 7 is

$$\begin{aligned}(C_{\Gamma 1}, M_{\Gamma 1}) &= (C_{\theta 1}, M_{\theta 1}) \otimes (C_{\theta 2}, M_{\theta 2}) \\ &= (P_{\Gamma 1}, T_{\Gamma 1}, F_{\Gamma 1}, M_{\Gamma 1})\end{aligned}$$

where $P_{\Gamma 1} = \{p_{\theta 1}, p_{\theta 2}\}$, $T_{\Gamma 1} = \{t_{11}, t_{12}, t_{13}, t_{21}, t_{22}, t_{23}\}$, $F_{\Gamma 1} = \{(p_{\theta 1}, t_{11}), (p_{\theta 1}, t_{21}), (t_{12}, p_{\theta 1}), (t_{23}, p_{\theta 1}), (p_{\theta 2}, t_{11}), (p_{\theta 2}, t_{21}), (t_{13}, p_{\theta 2}), (t_{22}, p_{\theta 2})\}$, and $M_{\Gamma 1}(p_{\theta 1}) = M_{\Gamma 1}(p_{\theta 2}) = 1$.

The controlled Petri net $(N, M_0) \otimes (C_{\Gamma 1}, M_{\Gamma 1})$ is shown in Fig. 2, and it is easy to check that it is live.

Let $(C_{\theta 3}, M_{\theta 3}) = (\{p_{\theta 3}\}, T_{\theta 3}, F_{\theta 3}, M_{\theta 3})$ be the Petri net controller for θ_3. By Definition 6, we have $M_{\theta 3}(p_{\theta 3}) = M_0(\Re[\theta_3]) - \xi_{\theta 3} = 3 - \xi_{\theta 3}$, where $\xi_{\theta 3}$ is with $1 \le \xi_{\theta 3} \le 2$. $I(\theta_3) = \{t_{11}, t_{21}\}$, $O(\theta_3) = \{t_{13}, t_{23}\}$, and $Z(\theta_3) = \emptyset$. $T_{\theta 3} = I(\theta_3) \cup O(\theta_3) \cup Z(\theta_3) = \{t_{11}, t_{13}, t_{21}, t_{23}\}$. $F_{\theta 3} = \{(p_{\theta 3}, t_{11}), (p_{\theta 3}, t_{21}), (t_{13}, p_{\theta 3}), (t_{23}, p_{\theta 3})\}$.

Then, $(C_{\theta 3}, M_{\theta 3})$ is the Petri net controller with respect to $\Gamma_2 = \{\theta_3\}$.

In $(C_{\theta 3}, M_{\theta 3})$, $1 \le \xi_{\theta 3} \le 2$. It can be checked that if $\xi_{\theta 3} = 2$, then the controlled Petri net $(N, M_0) \otimes (C_{\theta 3}, M_{\theta 3})$ is live and shown in Fig. 3. If $\xi_{\theta 3} = 1$, $(N, M_0) \otimes (C_{\theta 3}, M_{\theta 3})$ is not live because $M = 4p_{10} + 4p_{20} + p_{12} + p_{21} + r_1$ is a reachable marking of $(N, M_0) \otimes (C_{\theta 3}, M_{\theta 3})$; t_{13} and t_{22} are process-enabled but dead at M.

From the definition of a Petri net controller with respect to a transition cover, we know that the controller structure is uniquely determined by the elements of transition covers, whereas its initial marking is dependent on control variables. Here are two questions for designing a Petri net controller based on a transition cover. One is, for a given transition cover, whether or not a solution for control variables ξ_θ exists such that the controlled system is live, and when it exists, how to determine them. The other is for what kind of transition covers the Petri net controller has a desired permissive and simple structure. In the next section, we discuss those questions.

IV. Transition-Cover-Based Petri Net Controller for S^{3}PR

Here, the determination of control variables is first discussed. A transition cover is then computed. Finally, a method is presented for computing an effective transition cover for which a Petri net controller exists.

A. Determination of Control Variables for Deadlock Control

By Lemma 1, the Petri net controller (C_Γ, M_Γ) with $\xi_\alpha = 1$, $\forall \alpha \in \Gamma$, can prevent each $\alpha \in \Gamma$ from being saturated at any reachable marking $M \in R(CN_\Gamma, M_{\Gamma 0})$, but cannot necessarily prevent all MPCs in $\Theta \setminus \Gamma$ from being so. Then, to prevent deadlocks in $(CN_\Gamma, M_{\Gamma 0})$, the control variables must be properly chosen.

In what follows, we will establish the conditions under which every MPC in $\Theta \setminus \Gamma$ is not saturated in $(CN_\Gamma, M_{\Gamma 0})$. We first give some notations used throughout this paper.

Let $\Gamma \subseteq \Theta$, $\alpha \in \Gamma$, and Δ_α denote the set of operation places that are previous to $\Im[\alpha]$, i.e., $\Delta_\alpha = \{p \in P | p < \Im[\alpha]\}$. Let $A_\alpha = \Delta_\alpha \setminus {}^{(o)}\Im[\alpha]$ and $A_\Gamma = \cup_{\alpha \in \Gamma} A_\alpha$.

Lemma 2: Let Γ be a minimal transition cover of $\theta \in \Theta \setminus \Gamma$ and $M \in R(CN_\Gamma, M_{\Gamma 0})$; θ cannot be saturated at M if

$$\forall \alpha \in \Gamma, 1 \le \xi_\alpha \le M_0(\Re[\alpha]) - 1 \tag{1}$$

$$\begin{aligned}\Sigma_{\alpha \in \Gamma} \xi_\alpha \ge \Sigma_{\alpha \in \Gamma} [M_0(\Re[\alpha]) - M(A_\alpha)] \\ - M_0(\Re[\theta]) + 1.\end{aligned} \tag{2}$$

Proof: By the definition of controller (C_Γ, M_Γ), we have that for each $\alpha \in \Gamma$, $M(\Delta_\alpha) = M({}^{(o)}\Im[\alpha]) + M(A_\alpha) \le$

$M_0(\Re[\alpha]) - \xi_\alpha$. Thus, for each $\alpha \in \Gamma$, $M(^{(o)}\Im[\alpha]) \leq M_0(\Re[\alpha]) - \xi_\alpha - M(A_\alpha)$. $^{(o)}\Im[\theta] \subseteq {}^{(o)}\Im[\Gamma] = \cup_{\alpha\in\Gamma} {}^{(o)}\Im[\alpha]$ because Γ is a transition cover of θ. Hence, $M(^{(o)}\Im[\theta]) \leq \Sigma_{\alpha\in\Gamma} M(^{(o)}\Im[\alpha]) \leq \Sigma_{\alpha\in\Gamma}[M_0(\Re[\alpha]) - \xi_\alpha - M(A_\alpha)]$. By (2), we have $\Sigma_{\alpha\in\Gamma}[M_0(\Re[\alpha]) - \xi_\alpha - M(A_\alpha)] \leq M_0(\Re[\theta]) - 1 < M_0(\Re[\theta])$, and $M(^{(o)}\Im[\theta]) < M_0(\Re[\theta])$. That is, θ cannot be saturated at M. ■

Corollary 1: Let Γ be a minimal transition cover of $\theta \in \Theta \setminus \Gamma$ satisfying (1) and $M \in R(CN_\Gamma, M_{\Gamma 0})$. Then, θ cannot be saturated at M if

$$\Sigma_{\alpha\in\Gamma}\xi_\alpha \geq \Sigma_{\alpha\in\Gamma} M_0\left(\Re[\alpha]\right) - M_0\left(\Re[\theta]\right) - M(A_\Gamma) + 1. \quad (3)$$

Proof: Since $\Sigma_{\alpha\in\Gamma} M(A_\alpha) \geq M(A_\Gamma)$, inequality (3) implies (2). ■

Let Γ be a minimal transition cover of θ. Denote $B_\theta = \{p \in A_\Gamma \cap {}^{(o)}\Im[\theta] | R(p) = r$ and there is no $q \in H(r)$ such that $q \in {}^{(o)}\Im[\theta] \setminus A_\Gamma\}$ and $k_\theta = M_0(R(B_\theta))$.

If θ is saturated at $M \in R(CN_\Gamma, M_{\Gamma 0})$, then $M(^{(o)}\Im[\theta]) = M_0(\Re[\theta])$ and $M(B_\theta) = M_0(R(B_\theta))$.

Lemma 3: Let Γ be a minimal transition cover of $\theta \in \Theta \setminus \Gamma$ satisfying (1). Then, θ cannot be saturated at any $M \in R(CN_\Gamma, M_{\Gamma 0})$ if

$$\Sigma_{\alpha\in\Gamma}\xi_\alpha \geq \Sigma_{\alpha\in\Gamma} M_0\left(\Re[\alpha]\right) - M_0\left(\Re[\theta]\right) - k_\theta + 1. \quad (4)$$

Proof: Let $\wp_1 = \{M_1 \in R(CN_\Gamma, M_{\Gamma 0}) | M_1(A_\Gamma) < k_\theta\}$ and $\wp_2 = R(CN_\Gamma, M_{\Gamma 0}) \setminus \wp_1$.

Let $M \in R(CN_\Gamma, M_{\Gamma 0})$. If $M \in \wp_1$, then $M(A_\Gamma) < k_\theta$. Since $B_\theta \subseteq A_\Gamma$, $M(B_\theta) \leq M(A_\Gamma) < k_\theta = M_0(R(B_\theta))$. Here, we can claim that θ is not saturated at M. To see this, assume, on the contrary, that θ is saturated at M. Then, $M(^{(o)}\Im[\theta]) = M_0(\Re[\theta])$, and $M(B_\theta) = M_0(R(B_\theta))$. This contradicts $M(B_\theta) < M_0(R(B_\theta))$.

If $M \in \wp_2$, then $M(A_\Gamma) \geq k_\theta$, and hence, $\Sigma_{\alpha\in\Gamma} M_0(\Re[\alpha]) - M_0(\Re[\theta]) - k_\theta \geq \Sigma_{\alpha\in\Gamma} M_0(\Re[\alpha]) - M_0(\Re[\theta]) - M(A_\Gamma)$. By (4) and Corollary 1, θ cannot be saturated at M.

Hence, θ cannot be saturated at any reachable marking M of $(CN_\Gamma, M_{\Gamma 0})$. ■

Lemma 4: Let Γ be a minimal transition cover of $\theta \in \Theta \setminus \Gamma$. There exists ξ_α such that both (1) and (4) hold if

$$M_0\left(\Re[\theta]\right) > |\Gamma|. \quad (5)$$

Proof: Let $\alpha \in \Gamma$ and $\xi_\alpha = M_0(\Re[\alpha]) - 1$. Since α has at least two resource places by the property of S^{3}PRs, $M_0(\Re[\alpha]) \geq 2$. By considering $\xi_\alpha \geq 1$, we have that (1) holds. Since $M_0(\Re[\theta]) > |\Gamma|$ and $k_\theta \geq 0$, $\Sigma_{\alpha\in\Gamma}\xi_\alpha = \Sigma_{\alpha\in\Gamma} M_0(\Re[\alpha]) - |\Gamma| > \Sigma_{\alpha\in\Gamma} M_0(\Re[\alpha]) - M_0(\Re[\theta]) \geq \Sigma_{\alpha\in\Gamma} M_0(\Re[\alpha]) - M_0(\Re[\theta]) - k_\theta$. Since ξ_α is a positive integer, $\Sigma_{\alpha\in\Gamma}\xi_\alpha \geq \Sigma_{\alpha\in\Gamma} M_0(\Re[\alpha]) - M_0(\Re[\theta]) - k_\theta + 1$, i.e., (4) holds. ■

Definition 8: Let $\Gamma \subseteq \Theta$ be a minimal transition cover of $\theta \in \Theta \setminus \Gamma$. Γ is called an effective transition cover of θ if Γ satisfies (5).

Let $\Gamma \subseteq \Theta$ be a transition cover of N. If $\forall \theta \in \Theta \setminus \Gamma$ there exists a subset $\Gamma(\theta) \subseteq \Gamma$ that is an effective transition cover of θ, then Γ is called an effective transition cover of (N, M_0).

The value $\xi_\alpha = M_0(\Re[\alpha]) - 1$ given in Lemma 4 is the largest satisfying (1). However, to obtain the maximal permissiveness of the controlled system, ξ_α should be chosen as small as possible. The smaller ξ_α, the more reachable markings of $(CN_\Gamma, M_{\Gamma 0})$ can reach and, thus, the better its performance.

The previous analysis leads to the following LIP problem:

$$\text{LIP1} : \text{Min } \Sigma_{\alpha\in\Gamma}\xi_\alpha$$
$$\text{s.t. Constraints (1) and (4)}, \xi_\alpha \in \mathbb{Z}^+$$

where Γ is an effective transition cover of (N, M_0).

Lemma 5: LIP1 is solvable.

Proof: $\forall \theta \in \Theta \setminus \Gamma$, let $\Gamma(\theta) \subseteq \Gamma$ be an effective transition cover of θ. Then, by Definition 8, $M_0(\Re[\theta]) > |\Gamma(\theta)|$. By Lemma 4, there exists $\xi_\alpha(\theta)$, $\alpha \in \Gamma(\theta)$, such that constraints (1) and (4) hold. Let $\xi_\alpha = \max\{\xi_\alpha(\theta) | \alpha \in \Gamma(\theta), \theta \in \Theta \setminus \Gamma\}$. Then, $\{\xi_\alpha | \alpha \in \Gamma\}$ is a solution for LIP1. ■

Theorem 1: Let Γ be an effective transition cover of (N, M_0). (C_Γ, M_Γ) is the Petri net controller for Γ, and its control variables are obtained by solving LIP1. Then

1) any MPC of N cannot be saturated at any reachable marking of $(CN_\Gamma, M_{\Gamma 0})$; and
2) $(CN_\Gamma, M_{\Gamma 0})$ is live.

Proof:

1) Let $\theta \in \Theta$. If $\theta \in \Gamma$, θ cannot be saturated in $(CN_\Gamma, M_{\Gamma 0})$ by Lemma 1.

If $\theta \in \Theta \setminus \Gamma$, then there exists an effective transition cover $\Gamma(\theta) \subseteq \Gamma$ of θ and $M_0(\Re[\theta]) > |\Gamma(\theta)|$ by Definition 8. By Lemma 4, there exists ξ_α, $\alpha \in \Gamma(\theta)$, such that both (1) and (4) hold. Hence, θ cannot be saturated in $(CN_\Gamma, M_{\Gamma 0})$ by Lemma 3.

From the preceding analysis, Conclusion 1 is proved.

2) Assume, on the contrary, that $(CN_\Gamma, M_{\Gamma 0})$ is not live. Then, $\exists M \in R(CN_\Gamma, M_{\Gamma 0})$, such that $D(CN_\Gamma, M) \neq \emptyset$. Any transition in $D(CN_\Gamma, M)$ is process-enabled but not resource or not control-enabled at M.

If all transitions in $D(CN_\Gamma, M)$ are process-enabled but not resource-enabled, let $t_1 \in D(CN_\Gamma, M)$. Then, t_1 is process-enabled but not resource-enabled at M, i.e., $M(^{(o)}t_1) > 0$ and $M(^{(r)}t_1) = 0$. Let $r_1 = {}^{(r)}t_1$, then $K(r_1) = \{p \in P | p \in H(r_1), M(p) > 0\}$ is not empty, $M(K(r_1)) = M_0(r_1)$, and $K(r_1)^\bullet \subseteq D(CN_\Gamma, M)$. For transition $t_2 \in K(r_1)^\bullet$, do the same aforementioned analysis as for t_1. Let $r_2 = {}^{(r)}t_2$, then $r_1 \neq r_2$, $M(r_2) = 0$ and $K(r_2) = \{p \in P | p \in H(r_2), M(p) > 0\}$ is not empty, $M(K(r_2)) = M_0(r_2)$, and $K(r_2)^\bullet \subseteq D(CN_\Gamma, M)$. Repeating the aforementioned analysis, a sequence of resources $r_1, r_2, \ldots$ is obtained. By the finiteness of the resource set P_R, there must exist integers u and k such that $r_u = r_{u+k}$. Let $R_1 = \{r_{u+i} | i = 1, 2, \ldots, k\}$ and $T_1 = \{t \in T | M(^{(o)}t) > 0, t^{(r)} \in R_1\}$. Then, from the definition of R_1 and T_1, T_1 satisfies $M(^{(o)}T_1) = M_0(R_1)$. Each $t \in T_1$ is process-enabled but not resource-enabled at M. Let $T_2 = \{t \in T_1 | {}^{(r)}t \notin R_1\}$. If $T_2 \neq \emptyset$, repeat the following procedure till $T_2 = \emptyset$. $\forall t_0 \in T_2$, let $r = {}^{(r)}t_0$ and $Y(r) = \{t \in T | t^{(r)} = r, M(^{(o)}t) > 0\}$, then $M(^{(o)}Y(r)) = M_0(r)$. Set $R_1 = R_1 \cup \{r\}$, $T_1 = T_1 \cup Y(r)$, and $T_2 = \{t \in T_1 | {}^{(r)}t \notin R_1\}$. In each repetition of the aforementioned procedure, at least one transition is moved from T_2 into T_1. By the finiteness of T, the aforementioned procedure can stop in finite steps, and after it stops, $T_2 = \emptyset$. Finally, a set of resources R_1 and a set of transitions T_1 are obtained such that $\forall t \in T_1,\ M({}^{(o)}t) > 0$, $M({}^{(r)}t) = 0$, and ${}^{(r)}t, t^{(r)} \in R_1$. Since operations in ${}^{(o)}T_1$ occupy all resources in R_1, $M({}^{(o)}T_1) = M_0(R_1)$ and $R({}^{(o)}T_1) = R_1$. Let $N[P_1, T_1]$ denote the subnet of the S^{3}PR generated by P_1 and T_1. If $N[P_1, T_1]$ is strongly connected, then it is a PRTC with resource set R_1 and saturated at M. Otherwise, $N[P_1, T_1]$ consists of some strongly connected subnets and some directed paths between them. Then, there must exist, at least, a strongly connected subnet, which is denoted by N', such that from N' to any other strongly connected subsets, there exist no directed paths. Hence, N' is a PRTC with resource set R_1 and saturated at M. Since any PRTC with resource set R_1 is a subcircuit of the MPC with resource set R_1, there is an MPC with resource set R_1 that is saturated at M. However, this contradicts Conclusion 1.

Thus, there exists at least one transition $t_3 \in D(CN_\Gamma, M)$, such that t_3 is process and resource-enabled but not control-enabled. By Definition 6, $t_3 \in P_0^\bullet$. Consequently, $\exists p_c \in {}^{(c)}t_3$, such that $M(p_c) = 0$, and every transition $t_4 \in {}^\bullet p_c$ is dead at M. By Definition 6, ${}^{(c)}t_4 = \emptyset$. Hence, t_4 is process-enabled but not resource-enabled at M. Again, similar to the aforementioned analysis, there exists an MPC $\theta \in \Theta$, such that θ is saturated at M, which is in contradiction with Conclusion 1. Thus, $(CN_\Gamma, M_{\Gamma 0})$ is live. ■

In Example 2, $\Gamma_1 = \{\theta_1, \theta_2\}$ is an effective transition cover of θ_3 because $M_0(\Re[\theta_3]) > |\Gamma_1|$. Thus, it is also an effective transition cover of (N, M_0) shown in Fig. 1. For θ_1, $\Delta_{\theta 1} = \{p \in P | p < \Im[\theta_1]\} = \{p_{11}, p_{21}, p_{22}\}$ and $A_{\theta 1} = \Delta_{\theta 1} \setminus {}^{(o)}\Im[\theta_1] = \{p_{21}\}$. For θ_2, $\Delta_{\theta 2} = \{p_{11}, p_{12}, p_{21}\}$ and $A_{\theta 2} = \{p_{11}\}$. Hence, $A_{\Gamma 1} = A_{\theta 1} \cup A_{\theta 2} = \{p_{11}, p_{21}\}$. $A_{\Gamma 1} \cap {}^{(o)}\Im[\theta_3] = \{p_{11}, p_{21}\} \cap \{p_{11}, p_{12}, p_{21}, p_{22}\} = \{p_{11}, p_{21}\}$. Since $R(p_{11}) = r_1$, $H(r_1) = \{p_{11}, p_{23}\}$, $p_{23} \notin {}^{(o)}\Im[\theta_3]$, and $p_{11} \in B_{\theta 3}$. Similarly, we have $p_{21} \in B_{\theta 3}$. Thus, $B_{\theta 3} = \{p_{11}, p_{21}\}$, and $k_{\theta 3} = M_0(R(B_{\theta 3})) = M_0(r_1) + M_0(r_3) = 2$.

To obtain a live Petri net controller for (N, M_0) with respect to Γ_1, we need to solve the following LIP:

$$\begin{aligned}
\text{LIP2}: &\ \text{Min } \xi_1 + \xi_2 \\
&\quad \text{s.t. } 1 \le \xi_1 \le M_0(\Re[\theta_1]) - 1 = 1 \\
&\quad 1 \le \xi_2 \le M_0(\Re[\theta_2]) - 1 = 1 \\
&\quad \xi_1 + \xi_2 \ge M_0(\Re[\theta_1]) + M_0(\Re[\theta_2]) - M_0(\Re[\theta_3]) \\
&\qquad - k_{\theta 3} + 1 = 0 \\
&\quad \xi_1, \xi_2 \in \mathbb{Z}^+.
\end{aligned}$$

LIP2 has a solution $\xi_1 = \xi_2 = 1$. By Theorem 1, the obtained controlled Petri net shown in Fig. 2 is live.

For an ineffective transition cover, the corresponding LIP may not have a solution. In this case, we cannot obtain a live Petri net controller by the preceding method. Consider the following example.

Example 3: Consider a marked S^{3}PR (N, M_0) shown in Fig. 4. Its resource-transition net is shown in Fig. 5. $\Theta = \{\theta_1, \theta_2, \theta_3, \theta_4\}$, where $\theta_1 = m_1 t_{42} m_2 t_{32} m_1$, $\theta_2 = m_2 t_{52} m_3 t_{22} m_2$, $\theta_3 = m_1 t_{62} m_3 t_{12} m_1$, and $\theta_4 = m_1 t_{62} m_3 t_{22} m_2 t_{32} m_1 t_{42} m_2 t_{52} m_3 t_{12} m_1$. Let $\Gamma = \{\theta_1, \theta_2, \theta_3\}$. Then, Γ is a transition cover of N and a minimal one of θ_4.

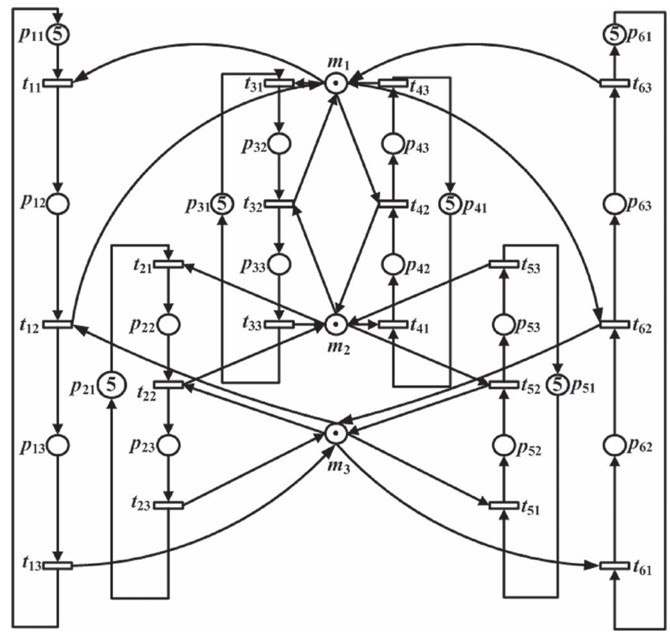


Fig. 4. Marked S^{3}PR (N, M_0).

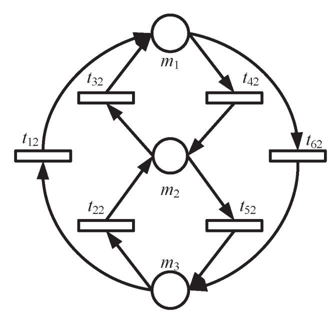


Fig. 5. Resource-transition net of (N, M_0) shown in Fig. 4.

For Γ and θ_4, $M_0(\Re[\theta_4]) = |\Gamma| = 3$ and (5) does not hold. Since no other subsets of Γ can cover θ_4, Γ is an ineffective transition cover of θ_4, and hence, Γ is an ineffective one of (N, M_0).

Since $\Delta_{\theta 1} = {}^{(o)}\Im[\theta_1] = \{p_{32}, p_{42}\}$, $A_{\theta 1} = \emptyset$. Similarly, $A_{\theta 2} = A_{\theta 3} = \emptyset$. Hence, $A_\Gamma = B_{\theta 4} = \emptyset$, and $k_{\theta 4} = 0$.

The LIP with respect to Γ is as follows:

$$\begin{aligned}
\text{LIP3}: &\ \text{Min } \xi_1 + \xi_2 + \xi_3 \\
&\quad \text{s.t. } 1 \le \xi_1 \le M_0(\Re[\theta_1]) - 1 = 1 \\
&\quad 1 \le \xi_2 \le M_0(\Re[\theta_2]) - 1 = 1 \\
&\quad 1 \le \xi_3 \le M_0(\Re[\theta_3]) - 1 = 1 \\
&\quad \xi_1 + \xi_2 + \xi_3 \ge M_0(\Re[\theta_1]) + M_0(\Re[\theta_2]) \\
&\qquad + M_0(\Re[\theta_3]) - M_0(\Re[\theta_4]) \\
&\qquad - k_{\theta 4} + 1 = 4 \\
&\quad \xi_1,\ \xi_2,\ \xi_3 \in \mathbb{Z}^+.
\end{aligned}$$

TABLE I
PETRI NET CONTROLLER (C_Γ, M_Γ) TO Γ OF (N, M_0) SHOWN IN FIG. 4

p_θ	${}^\bullet p_\theta$	$p_\theta^\bullet$	$M_\Gamma(p_\theta)$
$p_{\theta 1}$	t_{32}, t_{42}	t_{31}, t_{41}	1
$p_{\theta 2}$	t_{22}, t_{52}	t_{21}, t_{51}	1
$p_{\theta 3}$	t_{12}, t_{62}	t_{11}, t_{61}	1

No choice of ξ_1–ξ_3 satisfies the preceding restrictions; hence, LIP3 has no solution. Thus, we cannot design a live Petri net controller for the system by solving LIP3. If we only control θ_1–θ_3 by designing a Petri net controller given in Definition 7, the control variables must be $\xi_1 = \xi_2 = \xi_3 = 1$, and the controller (C_Γ, M_Γ) of (N, M_0) is displayed in Table I. It can be checked that the controlled Petri net $(CN_\Gamma, M_{\Gamma 0})$ is not live. The reason is that $M = 5p_{11} + 4p_{21} + p_{22} + 4p_{31} + p_{32} + 5p_{41} + 5p_{51} + 4p_{61} + p_{62}$ is a reachable marking of $(CN_\Gamma, M_{\Gamma 0})$, whereas θ_4 is saturated at it. Hence, $(CN_\Gamma, M_{\Gamma 0})$ is not live.

From [28], we know that there is a one-to-one correspondence between MPCs and SMSs. From all MPCs, i.e., $\theta_1 - \theta_4$, we can obtain four corresponding SMSs, namely, $S_1 = \{m_1, m_2, p_{12}, p_{22}, p_{33}, p_{43}, p_{53}, p_{63}\}$, $S_2 = \{m_2, m_3, p_{13}, p_{23}, p_{33}, p_{42}, p_{53}, p_{62}\}$, $S_3 = \{m_1, m_3, p_{13}, p_{23}, p_{32}, p_{43}, p_{52}, p_{63}\}$, and $S_4 = \{m_1, m_2, m_3, p_{13}, p_{23}, p_{33}, p_{43}, p_{53}, p_{63}\}$. It can be checked that $\{S_1, S_2, S_3\}$, corresponding to $\Gamma = \{\theta_1, \theta_2, \theta_3\}$, is a set of elementary siphons; and S_4 is strongly dependent on S_1, S_2, and S_3.

According to the deadlock prevention method based on elementary siphons proposed in [13], by adding a control place with a proper number of initial tokens for each elementary siphon, one can obtain a live Petri net controller for S^{3}PRs. For this example, the structures of controllers are the same for $\Gamma = \{\theta_1, \theta_2, \theta_3\}$ and for $\{S_1, S_2, S_3\}$ in [13], whereas our control variables are the control depth variables in [13]. Since the control (depth) variables must be $\xi_1 = \xi_2 = \xi_3 = 1$, by the method in [13], one should just obtain the Petri net controller as displayed in Table I, whereas this Petri net controller cannot guarantee the liveness of the controlled system.

From this example, one can see that the method in [13] for designing a live Petri net controller cannot adapt to any set of elementary siphons; instead, they must be properly chosen, as shown later.

B. Computation of Effective Transition Covers

In Section IV-A, a method is presented for designing a live Petri net controller for S^{3}PRs based on an effective transition cover by solving LIP1, where an effective transition cover plays an important role. Here, an algorithm for computing a transition cover, which may be not effective, and a method for transforming an ineffective one to an effective one are presented.

1) Computing Transition Covers: First, we present a property of a minimal circuit basis of a digraph [7].

Lemma 6: Let Θ be the set of all MPCs of N and Ξ be a minimal circuit basis of N_R that is the resource-transition net of N. For each $t \in \Im[\Theta]$, there exists an elementary RTC $\alpha \in \Xi$ containing t, and hence, $\Im[\Theta] \subseteq \Im[\Xi]$.

Proof: $\forall t \in \Im[\Theta]$, there exists an MPC $\theta \in \Theta$ such that $t \in \Im[\theta]$ and then an elementary RTC α in N_R such that $\alpha \subseteq \theta$ and $t \in \Im[\alpha]$. By the definition of minimal circuit basis, there exist some elementary RTCs $\alpha_1, \alpha_2, \ldots, \alpha_n \in \Xi$ such that vector C can be expressed as a linear combination of vectors $C_1, C_2, \ldots, C_n$, where $C(C_i)$ is the arc-indexed vector of α (α_i), $i \in \mathbb{N}_n = \{1, 2, \ldots, n\}$. Let $r_1 = {}^{(r)}t$ and $r_2 = t^{(r)}$. By the definition of the arc-indexed vector, the coordinate $C((r_1, t)) = C((t, r_2)) = 1$. Then, there exists some α_i such that $C_i((r_1, t)) = C_i((t, r_2)) = 1$, that is, $r_1 t r_2$ is a part of α_i, i.e., α_i contains t. Hence, $\Im[\Theta] \subseteq \Im[\Xi]$. ■

Algorithm CTC (Computing a Transition Cover of N)
Input: N_R, the resource-transition net of $N = (P \cup P_0 \cup P_R, T, F)$;
Output: Γ, a transition cover of N;
Step 1: Compute a minimal circuit basis of N_R, denoted by Ξ, by Algorithm 1 R-Greedy in [7]. Let $\Gamma = \Xi_X = \emptyset$;
Step 2: $\forall t \in N_R$, find an elementary RTC in Ξ, denoted by $\kappa[t]$, which contains t. If $\kappa[t]$ does not exist, denote $\kappa[t] = \emptyset$;
Step 3: (Maximization) For each $\alpha \in \Xi$, if $\gamma(\alpha) = \langle \Re[\alpha], \Re[\alpha]^\bullet \cap {}^\bullet \Re[\alpha] \rangle$ is perfect, then add $\gamma(\alpha)$ into Γ and, from Ξ, delete α and all elementary RTCs with the same resource set $\Re[\alpha]$. Otherwise, add α into Ξ_X;
Step 4: (Perfection) For each $\alpha \in \Xi_X$, do {
  Set flag = true;
  Let $\wp$ be a set of transitions; and set $\wp = \emptyset$;
  Let $\zeta = \alpha$ and $T_P[\zeta] = \{t \in \Im[\gamma(\zeta)] | \ ({}^{(o)}t)^\bullet \not\subseteq \Im[\gamma(\zeta)]\}$;
  while $T_P[\zeta] \setminus \wp \neq \emptyset$ **do**
    Select $t \in T_P[\zeta] \setminus \wp$;
    if There exists a transition $t_1 \in ({}^{(o)}t)^\bullet \setminus \Im[\gamma(\zeta)]$ such that $\kappa(t_1) = \emptyset$ **then**
      Delete α from Ξ_X;
      Exit While();
    end if
    for all $t_1 \in ({}^{(o)}t)^\bullet \setminus \Im[\gamma(\zeta)]$ **do**
      $\zeta = \zeta \cup \kappa(t_1)$;
    end for
    Add t into $\wp$;
    $T_P[\zeta] = \{t \in \Im[\gamma(\zeta)] | ({}^{(o)}t)^\bullet \not\subseteq \Im[\gamma(\zeta)]\}$;
  end while
  if flag = true **then**
    Add $\gamma(\zeta)$ to Γ;
  end if
Step 5: Output Γ;

We explain the correctness of **Algorithm CTC** as follows.

Step 1 finds a minimal circuit basis Ξ of N_R by [7]. Let N_{R0} denote the underlying undirected graph of N_R; $c(N_{R0})$ denote the number of connected components of N_{R0}; and a and v denote the number of arcs and vertices of N_R, respectively. Then, the number n_Ξ of elementary RTCs in Ξ is $a - v + c(N_{R0})$ by [7]. Since $c(N_{R0}) < v$, $n_\Xi < a$. Let v_t be the number of transition vertices in N_R. One transition vertex corresponds to two arcs in N_R, leading to $a = 2v_t$. Let $|T|$ be the number of transition vertices of N, $v_t < |T|$. Hence, $a < 2|T|$, and $n_\Xi < 2|T|$.

By Lemma 6, $\Im[\Theta] \subseteq \Im[\Xi]$ in Step 1. Hence, by Steps 3 and 4, $\forall t \in \Im[\Theta]$, there exists an MPC in Γ that contains t because an MPC in Γ is derived from an elementary RTC in Ξ by maximization and perfection. Thus, we have $\Im[\Theta] \subseteq \Im[\Gamma]$ and conclude that Γ is a transition cover of N.

By Steps 3 and 4, the number of MPCs in Γ, which is denoted by $|\Gamma|$, is, at most, that of elementary RTCs in Ξ. That is, $|\Gamma| \leq n_\Xi$. Thus, $|\Gamma| < 2|T|$.

Second, we present the complexity of **Algorithm CTC**.

By [7], the complexity of Step 1 is $O(n_\Xi a^2 v)$. In Step 3, an elementary RTC α is deleted from Ξ if $\gamma(\alpha)$ is perfect; otherwise, α is added into Ξ_X. Hence, when Step 3 is finished, for all $\alpha \in \Xi_X$, $\gamma(\alpha)$ is not perfect. The time complexity of Step 3 is $O(n_\Xi)$. In Step 4, from each $\alpha \in \Xi_X$, an MPC $\gamma(\zeta)$ (if it exists) is recursively constructed by merging an elementary RTC $\kappa(t_1)$ into α each time, where $t_1 \in ({}^{(o)}t)^\bullet \setminus \Im[\gamma(\zeta)]$. Thus, the time complexity of Step 4 is $O(n_\Xi v_t)$. Hence, the complexity of **Algorithm CTC** is $O(n_\Xi a^2 v) + O(n_\Xi) + O(n_\Xi v_t)$. Since $n_\Xi < a$ and $v_t < v$, the complexity of **Algorithm CTC** is no more than $O(a^3 v)$. Let $|R|$ be the number of resource places in N, $v \leq |T| + |R|$. Note that $a < 2|T|$; thus, the complexity of **Algorithm CTC** is no more than $O(|T|^3(|T| + |R|))$.

2) Transforming Ineffective Transition Covers to Effective Ones: The transition cover output by **Algorithm CTC** may be not effective. Here, an algorithm for transforming ineffective transition covers to effective ones is presented.

Let us first explain the transformation idea by an example.

Reconsider the Petri net shown in Fig. 4. From Example 3, we know that $\Theta = \{\theta_1, \theta_2, \theta_3, \theta_4\}$, where $\theta_1 = m_1 t_{42} m_2 t_{32} m_1$, $\theta_2 = m_2 t_{52} m_3 t_{22} m_2$, $\theta_3 = m_1 t_{62} m_3 t_{12} m_1$, and $\theta_4 = m_1 t_{62} m_3 t_{22} m_2 t_{32} m_1 t_{42} m_2 t_{52} m_3 t_{12} m_1$. $\Gamma = \{\theta_1, \theta_2, \theta_3\}$ is transition cover of N and a unique transition cover of θ_4. However, Γ is not an effective transition cover of θ_4 because $M_0(\Re[\theta_4]) = 3 = |\Gamma|$ and $M_0(\Re[\theta_4]) > |\Gamma|$ does not hold.

Note that, in inequality $M_0(\Re[\theta_4]) > |\Gamma|$, $M_0(\Re[\theta_4])$ is constant and cannot be changed. $\Gamma = \{\theta_1, \theta_2, \theta_3\}$ is the only minimal transition cover of θ_4. $\Im[\theta_i] \cap \Im[\theta_4] \neq \emptyset$, $i \in \mathbb{N}_3 = \{1, 2, 3\}$. If replacing θ_3 by $\delta(\Re[\theta_3 \cup \theta_4]) = \theta_4$ in Γ, we can obtain a new transition cover $\Gamma_1 = \{\theta_1, \theta_2, \theta_4\}$. Then, in Γ_1, the minimal transition cover of θ_4 is $\Gamma_1(\theta_4) = \{\theta_4\}$, and $M_0(\Re[\theta_4]) = 3 > |\Gamma_1(\theta_4)| = 1$ holds. The minimal transition cover of θ_3 is $\Gamma_1(\theta_3) = \{\theta_4\}$, and $M_0(\Re[\theta_3]) = 2 > |\Gamma_1(\theta_3)| = 1$ holds. Now, we can claim that Γ_1 is an effective transition cover of (N, M_0) shown in Fig. 4.

The detailed algorithm is as presented here.

Algorithm TIE (Transforming an Ineffective Transition Cover to an Effective One)
Input: A transition cover Γ_0 of N output by **Algorithm CTC**.
Output: Γ, an effective transition cover of N; $\Omega = \{\Gamma(\theta) | \Gamma(\theta)$ is an effective transition cover for $\theta \in \Theta \setminus \Gamma\}$.
Set $\Gamma = \Gamma_0$; $\Phi = \Gamma_0$; $\Omega = \emptyset$;
while $\Theta \setminus \Phi \neq \emptyset$ **do**
  Choose $\theta \in \Theta \setminus \Phi$;
  $\Gamma(\theta) = \{\alpha \in \Gamma | \Im[\alpha] \cap \Im[\theta] \neq \emptyset\}$;
  Let $s = |\Gamma(\theta)|$;
  Sort $\Gamma(\theta) = \{\alpha_0, \alpha_1, \ldots, \alpha_{s-1}\}$ by size $|\Im[\alpha_i] \cap \Im[\theta]|$ in a descending order;
  for int $i = 0$; $i++$; $i < s$ **do**
    if $\Im[\theta_2] \subseteq \Im[\Gamma(\theta) \setminus \{\alpha_i\}]$ **then**
      Let $\Gamma(\theta) = \Gamma(\theta) \setminus \{\alpha_i\}$;
    end if
  end for // Find a minimal transition cover $\Gamma(\theta)$ of θ.
  if $M_0(\Re[\theta]) > |\Gamma(\theta)|$ **then**
    $\Phi = \Phi \cup \{\theta\}$;
    $\Omega = \Omega \cup \{\Gamma(\theta)\}$;
    Continue;
      // $\Gamma(\theta)$ is an effective transition cover of θ.
      else
    Choose $\varpi \in \Gamma(\theta)$, and $\beta = \delta(\Re[\varpi \cup \theta])$;
    $\Gamma = (\Gamma \setminus \{\varpi\}) \cup \{\beta\}$;
    for all $\Gamma(\varepsilon) \in \Omega$ **do**
      if $\varpi \in \Gamma(\varepsilon)$ **then**
        $\Gamma(\varepsilon) = (\Gamma(\varepsilon) \setminus \{\varpi\}) \cup \{\beta\}$;
        for all $\chi \in \Gamma(\varepsilon)$ **do**
          If $\Im[\Gamma(\varepsilon)] \subseteq \Im[\Gamma(\varepsilon) \setminus \{\chi\}]$ then $\{\Gamma(\varepsilon) = \Gamma(\varepsilon) \setminus \{\chi\}$;
        end for
        // ensuring that $\Gamma(\varepsilon)$ is a minimal transition cover of ε.
      end if
    end for
    $\Phi = \Phi \cup \{\varpi, \theta, \beta\}$; $\Gamma(\varpi) = \{\beta\}$; $\Gamma(\theta) = \{\beta\}$; $\Omega = \Omega \cup \{\Gamma(\varpi), \Gamma(\theta)\}$;
  end if
end while
Output Γ and Ω;

In **Algorithm TIE**, for each MPC $\theta \in \Theta \setminus \Gamma$, one of its minimal transition covers, i.e., $\Gamma(\theta)$, is found. If for some $\theta \in \Theta \setminus \Gamma$, $M_0(\Re[\theta]) > |\Gamma(\theta)|$ does not hold, the transition cover Γ is not effective and is transformed into a new one, i.e., $\Gamma' = (\Gamma \setminus \{\varpi\}) \cup \{\beta\}$, where $\beta = \delta(\Re[\varpi \cup \theta])$ is the MPC with resource set $\Re[\varpi \cup \theta]$. Since both θ and ϖ are MPCs, $\beta \neq \emptyset$, and θ and ϖ are subcircuits of β. Thus, $\{\beta\}$ is a minimal transition cover of θ and ϖ and added into Ω. According to the new transition cover, i.e., Γ', of N, the already found effective transition cover $\Gamma(\varepsilon)$ for ε (where $\Gamma(\varepsilon)$ is already in Ω) is changed into $\Gamma'(\varepsilon) = (\Gamma(\varepsilon) \setminus \{\varpi\}) \cup \{\beta\}$ if $\varpi \in \Gamma(\varepsilon)$ and $\Gamma'(\varepsilon) = \Gamma(\varepsilon)$ if $\varpi \notin \Gamma(\varepsilon)$. At the same time, it needs guarantee that $\Gamma'(\varepsilon)$ is a minimal transition cover of ε in Γ'. We also note that $\Gamma'(\varepsilon)$ is an effective transition cover of ε, i.e., $M_0(\Re[\varepsilon]) > |\Gamma'(\varepsilon)|$ for each $\varepsilon \in \Phi' = \Phi \cup \{\varpi, \theta, \beta\}$. The reason is explained as follows. Since for $\varepsilon \in \{\varpi, \theta, \beta\}$, $\Gamma'(\varepsilon) = \{\beta\}, M_0(\Re[\varepsilon]) \geq 2$ and $|\Gamma'(\varepsilon)| = 1$, $M_0(\Re[\varepsilon]) > |\Gamma'(\varepsilon)|$ holds; and for $\varepsilon \in \Phi$, $|\Gamma'(\varepsilon)| = |\Gamma(\varepsilon)|$, and $M_0(\Re[\varepsilon]) > |\Gamma(\varepsilon)| = |\Gamma'(\varepsilon)|$ because $\Gamma(\varepsilon)$ is an effective transition cover of ε in Γ. Thus, for each $\varepsilon \in \Phi'$, $\Gamma'(\varepsilon)$ is an effective transition cover of ϵ in Γ'.

For each repeat of while(), at least one MPC (θ or θ and β) is added into Φ; hence, while() can repeat, at most, $|\Theta \setminus \Gamma_0|$ times. When while() finishes, we obtain an effective transition cover Γ of N, and an effective transition cover, i.e., $\Gamma(\varepsilon)$, for each $\varepsilon \in \Theta \setminus \Gamma$.

As a conclusion, we have the following results.

Theorem 2: For any transition cover Γ_0 of N, **Algorithm TIE** can correctly output an effective transition cover, i.e., Γ, of N and an effective transition cover, i.e., $\Gamma(\varepsilon)$, for each $\varepsilon \in \Theta \setminus \Gamma$.

In addition, note that the number of MPCs in the effective transition cover Γ computed by **Algorithm TIE** is the same as that of Γ_0 computed by **Algorithm CTC**, i.e., $|\Gamma| < 2|T|$.

Since the number of MPCs in $\Theta \setminus \Gamma_0$ exponentially grows with the size of Petri nets, the complexity of **Algorithm TIE** is exponential.

3) Deadlock Control Method Based on Transition Covers: Based on the preceding discussion, a novel method for designing a deadlock prevention policy for S^{3}PRs based on a transition cover is presented as follows.

Procedure DPP (Designing Deadlock Prevention Policy Based on Transition Covers).

Given an S^{3}PR (N, M_0), its resource-transition net N_R and its MPC set Θ.

Step 1: Using **Algorithm CTC** to compute a transition cover of N, i.e., Γ_0. If Γ_0 is not effective, then use **Algorithm TIE** to convert Γ_0 into an effective transition cover Γ of N; else Γ_0 is effective, $\Gamma = \Gamma_0$.

Step 2: For Γ, construct Petri net controller (C_Γ, M_Γ) for (N, M_0) as in Definition 7. Its control variables ξ_α, $\alpha \in \Gamma$, are determined by solving LIP1 in Step 4.

Step 3: For each $\theta \in \Theta \setminus \Gamma$, according to its effective transition cover $\Gamma(\theta)$ output by **Algorithm TIE**, form the inequality as follows:

$$\Sigma_{\alpha \in \Gamma(\theta)} \xi_\alpha \geq \Sigma_{\alpha \in \Gamma(\theta)} M_0\left(\Re[\alpha]\right) - M_0\left(\Re[\theta]\right) - k_\theta + 1.$$

Step 4: Construct LIP1 as shown in Section IV-A and solve a set of values of ξ_α.

Step 5: Output (C_Γ, M_Γ).

Theorem 3: The Petri net controller (C_Γ, M_Γ) constructed by **Procedure DPP** is live.

Proof: It immediately follows from Theorems 1 and 2. ■

V. Examples

Example 4: It is shown that $\Gamma = \{\theta_1, \theta_2, \theta_3\}$ is not an effective transition cover of θ_4 in Example 3 and not an effective transition cover of (N, M_0) shown in Fig. 4. By **Algorithm TIE**, $\Gamma_1 = \{\theta_1, \theta_2, \theta_4\}$ is computed to be an effective cover of (N, M_0) shown in Fig. 4, and $\{\theta_4\}$ is the effective transition cover of θ_3 in Γ_1. Add control places $p_{\theta 1}$, $p_{\theta 2}$, and $p_{\theta 4}$ to θ_1, θ_2, and θ_4, respectively. Control variables ξ_1, ξ_2, and ξ_4 are to be determined as follows.

Since $\{\theta_4\}$ is the effective transition cover of θ_3 in Γ_1 and $A_{\theta 4} = \Delta_{\theta 4} \setminus {}^{(o)}\Im[\theta_4] = \emptyset$ due to $\Delta_{\theta 4} = {}^{(o)}\Im[\theta_4] = \{p_{12}, p_{22}, p_{32}, p_{42}, p_{52}, p_{62}\}$. Thus, $B_{\theta 3} = \emptyset$, and $k_{\theta 3} = 0$.

Then, for θ_3, the following constraint can be derived:

$$\xi_4 \geq M_0\left(\Re[\theta_4]\right) - M_0\left(\Re[\theta_3]\right) - k_{\theta 3} + 1 = 3 - 2 - 0 + 1 = 2.$$

TABLE II
Petri Net Controller $(C_{\Gamma 1}, M_{\Gamma 1})$ of (N, M_0) Shown in Fig. 4

p_θ	$^\bullet p_\theta$	$p_\theta^\bullet$	$M_{\Gamma_1}(p_\theta)$
$p_{\theta 1}$	t_{32}, t_{42}	t_{31}, t_{41}	1
$p_{\theta 2}$	t_{22}, t_{52}	t_{21}, t_{51}	1
$p_{\theta 4}$	t_{12}, t_{22}, t_{32}, t_{42}, t_{52}, t_{62}	t_{11}, t_{21}, t_{31}, t_{41}, t_{51}, t_{61}	1

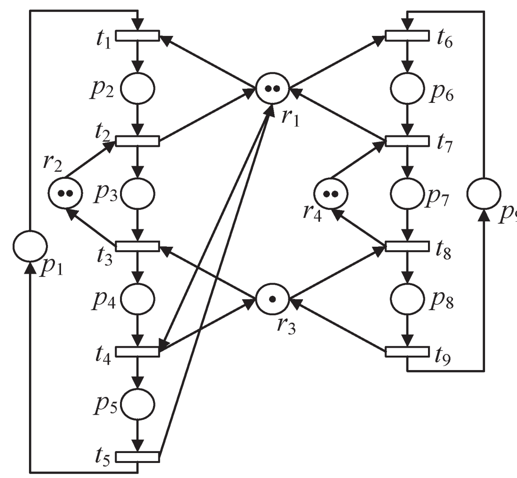


Fig. 6. Marked S^{3}PR (N, M_0).

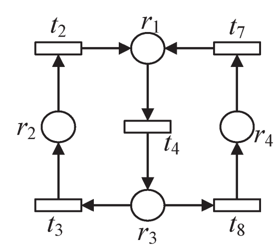


Fig. 7. Resource-transition net N_R.

Hence, the LIP is formed based on **Procedure DPP** as follows:

$$\begin{aligned} \text{LIP4} : \text{Min } & \xi_1 + \xi_2 + \xi_4 \\ \text{s.t. } & 1 \leq \xi_1 \leq 1 \\ & 1 \leq \xi_2 \leq 1 \\ & 1 \leq \xi_4 \leq 2 \\ & \xi_4 \geq 2 \\ & \xi_1, \xi_2, \xi_4 \in \mathbb{Z}^+. \end{aligned}$$

Obviously, the unique solution of LIP4 is $\xi_1 = \xi_2 = 1$, $\xi_4 = 2$. Thus, the Petri net controller $(C_{\Gamma 1}, M_{\Gamma 1})$ with respect to Γ_1 can be constructed, as shown in Table II. It is easy to check that the obtained Petri net controller is live.

Example 5: Consider a marked S^{3}PR (N, M_0) shown in Fig. 6. Its resource-transition net is shown in Fig. 7. By **Algorithm CTC**, we can obtain a transition cover $\Gamma = \{\theta_1, \theta_2\}$ of N, where $\theta_1 = r_1 t_4 r_3 t_3 r_2 t_2 r_1$ and $\theta_2 = r_1 t_4 r_3 t_8 r_4 t_7 r_1$. It is easy to compute that there are three MPCs of N, i.e., $\Theta = \{\theta_1, \theta_2, \theta_3\}$, where $\theta_3 = r_1 t_4 r_3 t_3 r_2 t_2 r_1 t_4 r_3 t_8 r_4 t_7 r_1$.

By **Algorithm TIE**, Γ is an effective transition cover of (N, M_0) shown in Fig. 6 and the effective transition cover of θ_3.

TABLE III
PETRI NET CONTROLLER (C_Γ, M_Γ) TO Γ OF (N, M_0) SHOWN IN FIG. 6

p_θ	${}^\bullet p_\theta$	$p_\theta^\bullet$	$M_\Gamma(p_\theta)$
$p_{\theta 1}$	t_4	t_1	4
$p_{\theta 2}$	t_4, t_8	t_1, t_6	4

TABLE IV
PETRI NET CONTROLLER (C_Π, M_Π) TO Π OF (N, M_0) SHOWN IN FIG. 6

p_S	${}^\bullet p_S$	$p_S^\bullet$	$M_\Pi(p_S)$
p_{S1}	t_4	t_1	4
p_{S2}	t_4, t_8	t_1, t_6	4
p_{S3}	t_4, t_8	t_1, t_6	6

By Definition 6, add control places $p_{\theta 1}$ and $p_{\theta 2}$ to θ_1 and θ_2, and denote the control variables ξ_1 and ξ_2, respectively. ξ_1 and ξ_2 satisfy $1 \le \xi_1 \le 4$ and $1 \le \xi_2 \le 4$.

For θ_3, its effective transition cover is $\Gamma(\theta_3) = \{\theta_1, \theta_2\}$. ${}^{(o)}\Im[\theta_1] = \{p_2, p_3, p_4\}$, $\Delta_{\theta 1} = \{p \in P | p < \Im[\theta_1]\} = \{p_2, p_3, p_4\}$, and $A_{\theta 1} = \Delta_{\theta 1} \setminus {}^{(o)}\Im[\theta_1] = \emptyset$. ${}^{(o)}\Im[\theta_2] = \{p_4, p_6, p_7\}$, and $\Delta_{\theta 2} = \{p \in P | p < \Im[\theta_2]\} = \{p_2, p_3, p_4, p_6, p_7\}$. Thus, $A_{\theta 2} = \Delta_{\theta 2} \setminus {}^{(o)}\Im[\theta_2] = \{p_2, p_3\}$. ${}^{(o)}\Im[\theta_3] = \{p_2, p_3, p_4, p_6, p_7\}$, and $(A_{\theta 1} \cup A_{\theta 2}) \cap {}^{(o)}\Im[\theta_3] = \{p_2, p_3\}$. Since $R(p_2) = r_1$, $H(r_1) = \{p_2, p_6\}$, and $p_6 \in {}^{(o)}\Im[\theta_3] \setminus (A_{\theta 1} \cup A_{\theta 2})$, we have $p_2 \notin B_{\theta 3}$. Similarly, $R(p_3) = r_2$, and $H(r_2) = \{p_3\}$; then, $p_3 \in B_{\theta 3}$. Hence, $B_{\theta 3} = \{p_3\}$, and $k_{\theta 3} = M_0(R(B_{\theta 3})) = 2$. Then, for θ_3, the following constraint can be derived:

$$\begin{aligned}\xi_1 + \xi_2 \ge M_0(\Re[\theta_1]) + M_0(\Re[\theta_2]) - M_0(\Re[\theta_3])\\ -k_{\theta 3} + 1 = 5 + 5 - 7 - 2 + 1 = 2.\end{aligned}$$

Hence, LIP5 is formed as follows:

$$\begin{aligned}\text{LIP5} : \text{Min } & \xi_1 + \xi_2\\ \text{s.t. } & 1 \le \xi_1 \le 4\\ & 1 \le \xi_2 \le 4\\ & \xi_1 + \xi_2 \ge 2\\ & \xi_1, \xi_2 \in \mathbb{Z}^+.\end{aligned}$$

The unique solution of LIP5 is $\xi_1 = \xi_2 = 1$. Thus, a live Petri net controller (C_Γ, M_Γ) can be constructed, as shown in Table III.

According to the one-to-one correspondence between SMSs and MPCs in [28], there are three SMSs corresponding to θ_1, θ_2, and θ_3 in N, respectively; where $S_1 = \{r_1, r_2, r_3, p_5, p_6, p_8\}$, $\{r_1, r_3, r_4, p_2, p_5, p_8\}$, and $\{r_1, r_2, r_3, r_4, p_5, p_8\}$. It is verified that $\Pi = \{S_1, S_2, S_3\}$ is a set of elementary siphons. By utilizing the deadlock prevention method based on elementary siphons proposed in [13], we can construct a live Petri net controller (C_Π, M_Π) shown in Table IV.

It can be checked that p_{S3} does not work in (C_Π, M_Π) and the performance of (C_Π, M_Π) is the same as that of (C_Γ, M_Γ).

Example 6: Consider a marked S^{3}PR (N, M_0) shown in Fig. 8, which was first presented by Ezpeleta *et al.* [4]. Its resource-transition net N_R [28] is shown in Fig. 9.

By Algorithm in [7], we can obtain one of minimal circuit bases $\Xi = \{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8\}$ of N_R, where $c_1 = p_{22} t_{18} p_{25} t_{12} p_{22}$, $c_2 = p_{25} t_{17} p_{23} t_{13} p_{25}$, $c_3 = p_{23} t_{16} p_{26} t_{14} p_{23}$, $c_4 = p_{25} t_3 p_{21} t_8 p_{25}$, $c_5 = p_{25} t_3 p_{21} t_2 p_{25}$, $c_6 = p_{20} t_6 p_{24} t_{19} p_{22} t_{18} p_{25} t_7 p_{20}$, $c_7 = p_{25} t_{17} p_{23} t_{16} p_{26} t_9 p_{21} t_8 p_{25}$, and $c_8 = p_{24} t_{19} p_{22} t_{11} p_{24}$. From Ξ, we can construct MPCs $\theta_i = c_i$, $i \in \mathbb{N}_3 = \{1, 2, 3\}$, $\theta_4 = c_4 \cup c_5$, $\theta_5 = c_1 \cup c_6 \cup c_8$, and $\theta_6 = c_2 \cup c_3 \cup c_4 \cup c_5 \cup c_7$, respectively. It can be checked that $\Gamma = \{\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6\}$ is a transition cover of N.

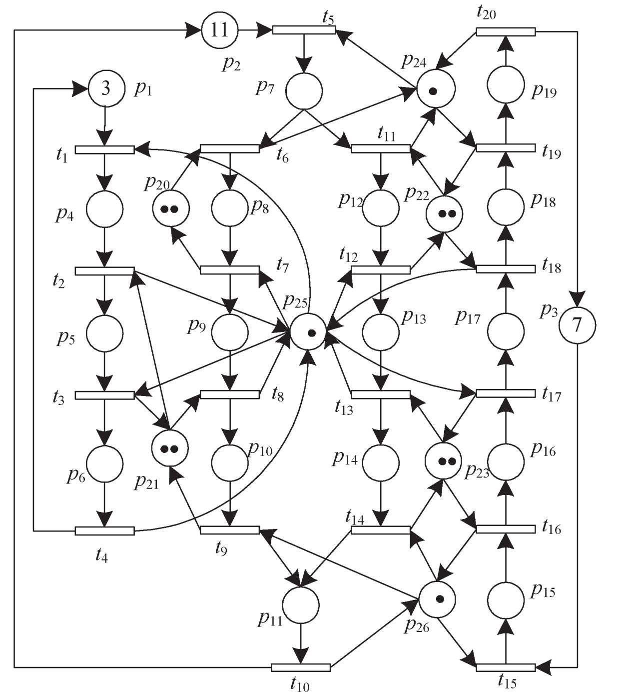


Fig. 8. Marked S^{3}PR (N, M_0).

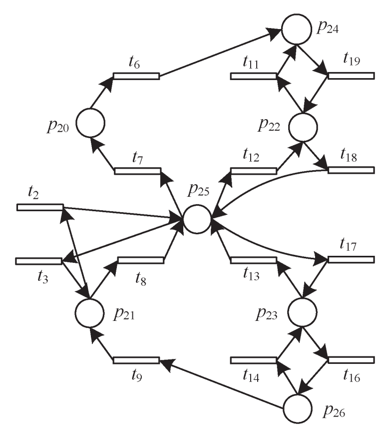


Fig. 9. Resource-transition net N_R.

The set of all MPCs in N is $\Theta = \Gamma \cup \{\theta_7, \theta_8, \ldots, \theta_{18}\}$, where $\theta_7 = \theta_1 \cup \theta_2$, $\theta_8 = \theta_1 \cup \theta_4$, $\theta_9 = \theta_1 \cup \theta_6$, $\theta_{10} = \theta_2 \cup \theta_3$, $\theta_{11} = \theta_2 \cup \theta_4$, $\theta_{12} = \theta_2 \cup \theta_5$, $\theta_{13} = \theta_4 \cup \theta_5$, $\theta_{14} = \theta_5 \cup \theta_6$, $\theta_{15} = \theta_1 \cup \theta_2 \cup \theta_3$, $\theta_{16} = \theta_1 \cup \theta_2 \cup \theta_4$, $\theta_{17} = \theta_2 \cup \theta_3 \cup \theta_5$, and $\theta_{18} = \theta_2 \cup \theta_4 \cup \theta_5$. It can be checked that Γ is an effective transition cover of (N, M_0) by **Algorithm TIE**.

By Definition 6, for each $\theta_i \in \Gamma$, add a control place $p_{\theta i}$ with control variable ξ_i. Control variables ξ_1–ξ_6 satisfy $1 \le \xi_1 \le 2$, $1 \le \xi_2 \le 2$, $1 \le \xi_3 \le 2$, $1 \le \xi_4 \le 2$, $1 \le \xi_5 \le 5$, and $1 \le \xi_6 \le 5$.

For θ_7, its effective transition cover, i.e., $\Gamma(\theta_7) = \{\theta_1, \theta_2\} \subseteq \Gamma$, can be obtained by **Algorithm TIE**. $\Delta_{\theta 1} = \{p \in$

TABLE V
PETRI NET CONTROLLER (C_Γ, M_Γ) OF (N, M_0) SHOWN IN FIG. 8

p_θ	$^\bullet p_\theta$	$p_\theta^\bullet$	$M_\Gamma(p_\theta)$
$p_{\theta 1}$	t_6, t_{12}, t_{18}	t_5, t_{15}	2
$p_{\theta 2}$	t_6, t_{13}, t_{17}	t_5, t_{15}	2
$p_{\theta 3}$	t_6, t_{14}, t_{16}	t_5, t_{15}	2
$p_{\theta 4}$	t_3, t_8, t_{11}	t_1, t_5	2
$p_{\theta 5}$	t_7, t_{12}, t_{19}	t_5, t_{15}	5
$p_{\theta 6}$	t_3, t_9, t_{14}, t_{17}	t_1, t_5, t_{15}	5

$P|p < \Im[\theta_1]\} = \{p_7, p_{12}, p_{15}, p_{16}, p_{17}\}$, $\Delta_{\theta 2} = \{p \in P|p < \Im[\theta_2]\} = \{p_7, p_{12}, p_{13}, p_{15}, p_{16}\}$, $A_{\theta 1} = \Delta_{\theta 1} \setminus {}^{(o)}\Im[\theta_1] = \{p_7, p_{15}, p_{16}\}$, and $A_{\theta 2} = \Delta_{\theta 2} \setminus {}^{(o)}\Im[\theta_2] = \{p_7, p_{12}, p_{15}\}$. ${}^{(o)}\Im[\theta_7] = \{p_{12}, p_{13}, p_{16}, p_{17}\}$, and $(A_{\theta 1} \cup A_{\theta 2}) \cap {}^{(o)}\Im[\theta_7] = \{p_{12}, p_{16}\}$. Since $R(p_{12}) = p_{22}$, $H(p_{22}) = \{p_{12}, p_{18}\}$, and $p_{18} \notin {}^{(o)}\Im[\theta_7]$, we have $p_{12} \in B_{\theta 7}$. Similarly, $p_{16} \in B_{\theta 7}$. Then, $B_{\theta 7} = \{p_{12}, p_{16}\}$. $M_0(p_{22}) = M_0(p_{23}) = 2$, and $k_{\theta 7} = 4$. Then, for θ_7, the following constraint can be derived:

$$\xi_1 + \xi_2 \geq M_0(\Re[\theta_1]) + M_0(\Re[\theta_2]) - M_0(\Re[\theta_7]) - k_{\theta 7} + 1 = -2.$$

Similarly, for θ_{15}, $\Gamma(\theta_{15}) = \{\theta_1, \theta_2, \theta_3\}$ is an effective transition cover of θ_{15} in Γ. $\Delta_{\theta 3} = \{p_7, p_{12}, p_{13}, p_{14}, p_{15}\}$, $A_{\theta 3} = \{p_7, p_{12}, p_{13}\}$, and ${}^{(o)}\Im[\theta_{15}] = \{p_{12}, p_{13}, p_{14}, p_{15}, p_{16}, p_{17}\}$; thus, $(A_{\theta 1} \cup A_{\theta 2} \cup A_{\theta 3}) \cap {}^{(o)}\Im[\theta_{15}] = \{p_{12}, p_{13}, p_{15}, p_{16}\}$. Since $H(p_{22}) = \{p_{12}, p_{18}\}$ and $p_{18} \notin {}^{(o)}\Im[\theta_{15}]$, we have $p_{12} \in B_{\theta 15}$. Repeating the aforementioned analysis, we have $p_{15} \in B_{\theta 15}, p_{13}, p_{16} \notin B_{\theta 15}$. Thus, $B_{\theta 15} = \{p_{12}, p_{15}\}$. $M_0(p_{22}) = 2$, $M_0(p_{26}) = 1$, and $k_{\theta 15} = 3$. For θ_5, the following constraint can be derived:

$$\xi_1 + \xi_2 + \xi_3 \geq M_0(\Re[\theta_1]) + M_0(\Re[\theta_2]) + M_0(\Re[\theta_3]) - M_0(\Re[\theta_{15}]) - k_{\theta 15} + 1 = 1.$$

Similarly, we can construct a constraint for each MPC in $\Theta \setminus \Gamma$ and then form the following LIP:

$$\begin{aligned}
\text{LIP6} : &\ \text{Min } \Sigma_{i=1}^{6} \xi_i \\
&\text{s.t. } 1 \leq \xi_i \leq 2, i = 1,2,3,4;\ 1 \leq \xi_i \leq 5, i = 5,6 \\
&\quad \xi_1 + \xi_2 \geq -2;\ \xi_1 + \xi_4 \geq 2;\ \xi_1 + \xi_6 \geq -1 \\
&\quad \xi_2 + \xi_3 \geq 1;\ \xi_2 + \xi_4 \geq 2;\ \xi_2 + \xi_5 \geq -1 \\
&\quad \xi_4 + \xi_5 \geq -1;\ \xi_5 + \xi_6 \geq -2 \\
&\quad \xi_1 + \xi_2 + \xi_3 \geq 1;\ \xi_1 + \xi_2 + \xi_4 \geq -1 \\
&\quad \xi_2 + \xi_3 + \xi_5 \geq 2;\ \xi_2 + \xi_4 + \xi_5 \geq -2 \\
&\quad \xi_i \in \mathbb{Z}^+, i = 1, 2, \ldots, 6.
\end{aligned}$$

By solving LIP6, its unique solution $\xi_1 = \xi_2 = \xi_3 = \xi_4 = \xi_5 = \xi_6 = 1$ is obtained. Then, using Γ and the solution of LIP6, the live Petri net controller (C_Γ, M_Γ) can be constructed, as shown in Table V. It can be checked that the performance of this controlled system is the same as the ones in [4] and [13].

VI. CONCLUSION

This paper has focused on the deadlock prevention problems in AMSs. Much existing work on deadlock prevention for Petri net models of AMSs has been based on siphons or RTCs. This paper has introduced, for the first time, the concept of transition covers, and based on it, a novel method is presented for designing a deadlock prevention policy for AMSs. An effective transition cover plays a key role in designing deadlock prevention policies, as elementary siphons in [13]. The number of MPCs in an effective transition cover is no more than twice that of transitions of Petri net models, which is much less than that of SMSs or MPCs for Petri net models, and in some cases, our examples show that it is not larger than the number of elementary siphons, and the performances of deadlock prevention policies based on elementary siphons and effective transition covers are the same.

A transition cover can be computed from a minimal circuit basis of a resource-transition net by **Algorithm CTC** that has polynomial time complexity. Not all transition covers can be used to construct a deadlock prevention policy. The example shows that, for an ineffective transition cover, which is also corresponding to a set of elementary siphons, by adding a control place to each MPC in the ineffective transition cover or the corresponding elementary siphon, we cannot obtain a live Petri net controller. In this paper, proposed **Algorithm TIE** can be used to check the effectiveness of a given transition cover. If it is not effective, then **Algorithm TIE** can transform it to an effective one. It is proved that, based on effective transition cover, a live Petri net controller for AMSs can be designed, whose control variables can be obtained by solving their corresponding LIP.

Our future research will include studying from what kind of effective transition covers, a better performance controller can be obtained, and extending our presented method to more general Petri nets.

REFERENCES

[1] C. Berge, *Graphs*. Amsterdam, The Netherlands: North Holland, 1985.

[2] Y. F. Chen, Z. W. Li, M. Khalgui, and O. Mosbahi, "Design of a maximally permissive liveness-enforcing Petri net supervisor for flexible manufacturing systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 2, pp. 374–393, Apr. 2011.

[3] F. Chu and X. Xie, "Deadlock analysis of Petri nets using siphons and mathematical programming," *IEEE Trans. Robot. Autom.*, vol. 13, no. 6, pp. 793–804, Dec. 1997.

[4] J. Ezpeleta, J. M. Colom, and J. Martinez, "A Petri net based deadlock prevention policy for flexible manufacturing systems," *IEEE Trans. Robot. Autom.*, vol. 11, no. 2, pp. 173–184, Apr. 1995.

[5] M. Fanti and M. C. Zhou, "Deadlock control methods in automated manufacturing systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 34, no. 1, pp. 5–22, Jan. 2004.

[6] M. Fanti, "Deadlock resolution strategy for automated manufacturing systems including conjunctive resource service," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 34, no. 1, pp. 80–92, Jan. 2004.

[7] P. M. Gleiss, J. Leydold, and P. F. Stadler, "Circuit bases of strongly connected digraphs," *Discuss. Math., Graph Theory*, vol. 23, no. 2, pp. 241–260, Jan. 2003.

[8] Y. S. Huang, M. D. Jeng, X. L. Xie, and S. L. Chung, "Deadlock prevention policy based on Petri nets and siphons," *Int. J. Prod. Res.*, vol. 39, no. 2, pp. 283–305, Jan. 2001.

[9] F. S. Hsieh, "Robustness analysis of holonic assembly/disassembly processes with Petri nets," *Automatica*, vol. 44, no. 10, pp. 2538–2548, Oct. 2008.

[10] M. D. Jeng, "A Petri net synthesis theory for modeling flexible manufacturing systems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 27, no. 2, pp. 169–183, Apr. 1997.

[11] M. D. Jeng and X. L. Xie, "Deadlock detection and prevention of automated manufacturing systems using Petri nets and siphons," in

Deadlock Resolution in Computer—Integrated Systems, M. Zhou and M. P. Fanti, Eds. New York: Marcel Dekker, 2005, pp. 233–281.

[12] T. Kumaran, W. Chang, and H. Cho, "A structured approach to deadlock detection, avoidance and resolution in flexible manufacturing systems," *Int. J. Prod. Res.*, vol. 32, no. 10, pp. 2361–2379, Oct. 1994.

[13] Z. W. Li and M. C. Zhou, "Elementary siphons of Petri nets and their application to deadlock prevention in flexible manufacturing systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 34, no. 1, pp. 38–51, Jan. 2004.

[14] Z. W. Li and M. C. Zhou, "Two-stage method for synthesizing liveness-enforcing supervisors for flexible manufacturing systems using Petri nets," *IEEE Trans Ind. Informat.*, vol. 2, no. 4, pp. 313–325, Nov. 2006.

[15] 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.

[16] Z. W. Li, N. Q. Wu, and M. C. Zhou, "Deadlock control of automated manufacturing systems based on Petri nets—A literature review," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 4, pp. 437–462, Jul. 2012.

[17] Z. W. Li and M. C. Zhou, *Deadlock Resolution in Automated Manufacturing Systems: A Novel Petri Net Approach*. New York, NY, USA: Springer, 2009.

[18] A. Nazeem, S. Reveliotis, Y. Wang, and S. Lafortune, "Designing compact and maximally permissive deadlock avoidance policies for complex resource allocation systems through classification theory: The linear case," *IEEE Trans. Autom. Control*, vol. 56, no. 8, pp. 1818–1833, Aug. 2011.

[19] L. Piroddi, R. Cordone, and I. Fumagalli, "Selective siphon control for deadlock prevention in Petri nets," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 38, no. 6, pp. 1337–1348, Nov. 2008.

[20] S. A. Reveliotis, M. A. Lawley, and P. M. Ferreira, "Polynomial-complexity deadlock avoidance policies for sequential resource allocation systems," *IEEE Trans. Autom. Control*, vol. 42, no. 10, pp. 1344–1357, Oct. 1997.

[21] N. Q. Wu, "Necessary and sufficient conditions for deadlock-free operation in flexible manufacturing systems using a colored Petri net model," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 29, no. 2, pp. 192–204, May 1999.

[22] N. Q. Wu and M. C. Zhou, "Avoiding deadlock and reducing starvation and blocking in automated manufacturing systems based on a Petri net model," *IEEE Trans. Robot. Autom.*, vol. 17, no. 5, pp. 658–699, Oct. 2001.

[23] N. Q. Wu and M. C. Zhou, "Modeling and deadlock avoidance of automated manufacturing systems with multiple automated guided vehicles," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 6, pp. 1193–1202, Dec. 2005.

[24] N. Q. Wu and M. C. Zhou, "Deadlock resolution in automated manufacturing systems with robots," *IEEE Trans. Autom. Sci. Eng.*, vol. 4, no. 3, pp. 474–480, Jul. 2007.

[25] N. Q. Wu, M. C. Zhou, and Z. W. Li, "Resource-oriented Petri net for deadlock avoidance in flexible assembly systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 38, no. 1, pp. 56–69, Jan. 2008.

[26] K. Y. Xing, B. S. Hu, and H. X. Chen, "Deadlock avoidance policy for Petri-net modeling of flexible manufacturing systems with shared resources," *IEEE Trans. Autom. Control*, vol. 41, no. 2, pp. 289–295, Feb. 1996.

[27] K. Y. Xing, M. C. Zhou, H. X. Liu, and F. Tian, "Optimal Petri-net-based polynomial-complexity deadlock-avoidance policies for automated manufacturing systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 39, no. 1, pp. 188–199, Jan. 2009.

[28] K. Y. Xing, M. C. Zhou, F. Wang, H. X. Liu, and F. Tian, "Resource-transition circuits and siphons for deadlock control of automated manufacturing systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 41, no. 1, pp. 74–84, Jan. 2011.

[29] K. Y. Xing, L. B. Han, M. C. Zhou, and F. Wang, "Deadlock-free genetic scheduling algorithm for automated manufacturing systems based on deadlock control policy," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 3, pp. 603–615, Jun. 2012.

[30] M. C. Zhou and M. P. Fanti, Eds., *Deadlock Resolution in Computer-Integrated Systems*. New York, NY, USA: Marcel Dekker, 2005.

[31] M. C. Zhou and K. Venkatesh, *Modeling, Simulation and Control of Flexible Manufacturing Systems: A Petri Net Approach*. Singapore: World Scientific, 1998.

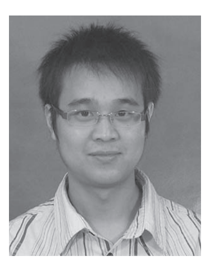
[32] M. C. Zhou and F. DiCesare, *Petri Net Synthesis for Discrete Event Control of Manufacturing Systems*. London: Kluwer Academic Publishers, 1993.

Huixia Liu received the B.S. degree in applied mathematics from Dalian University, Dalian, China, and the M.S. degree in mathematics from Harbin Institute of Technology, Harbin, China. She is currently working toward the Ph.D. degree in the Systems Engineering Institute, Xi'an Jiaotong University, Xi'an, China. She is also currently with the State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University. Her main research interests include control and scheduling of automated manufacturing and discrete event systems and Petri nets.

Keyi Xing (M'07) received the B.S. degree in mathematics from Northwest University, Xi'an, China, in 1982, the M.S. degree in applied mathematics from Xidian University, Xi'an, in 1985, and the Ph.D. degree in systems engineering from Xi'an Jiaotong University, Xi'an, in 1994. He was with Xidian University in 1985. Since 2004, he has been with Xi'an Jiaotong University, where he is currently a Professor of systems engineering with the State Key Laboratory for Manufacturing Systems Engineering and the Systems Engineering Institute. His research interests include control and scheduling of automated manufacturing, discrete event, and hybrid systems.

MengChu Zhou (S'88–M'90–SM'93–F'03) received the B.S. degree in electrical engineering from Nanjing University of Science and Technology, Nanjing, China, in 1983, the M.S. degree in automatic control from Beijing Institute of Technology, Beijing, China, in 1986, and the Ph.D. degree in computer and systems engineering from the Rensselaer Polytechnic Institute, Troy, NY, USA, in 1990.

In 1990, he joined New Jersey Institute of Technology (NJIT), Newark, NJ, USA, where he is a Professor of electrical and computer engineering. He is also a Professor with Tongji University, Shanghai, China. He has authored or coauthored over 480 publications, including 11 books, more than 220 journal papers (majority in IEEE transactions), and 17 book chapters. His research interests are in intelligent automation, Petri nets, sensor networks, semiconductor manufacturing, and energy systems.



Libin Han received the B.S. degree in automation science and technology from Xi'an Jiaotong University, Xi'an, China, in 2009, and is currently working toward the Ph.D. degree in systems engineering at the same university.

He is currently with the State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University. His main research interests include control and scheduling of automated manufacturing systems and Petri nets.

Feng Wang received the B.S. degree in applied mathematics from Northwest University, Xi'an, China, in 1997, and the M.S. degree in applied mathematics from Xi'an Jiaotong University, Xi'an, in 2004, where she is currently working toward the Ph.D. degree in the Systems Engineering Institute.

She is also currently with the State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University. Her main research interests include control and scheduling of manufacturing and discrete event systems.