Special Issue Article

Advances in Mechanical Engineering

Advances in Mechanical Engineering
2017, Vol. 9(11) 1–14
© The Author(s) 2017
DOI: 10.1177/1687814017734709
journals.sagepub.com/home/ade
SAGE

An elementary siphon-based deadlock control algorithm with maximally reachable number to cope with deadlock problems in ordinary Petri nets

Shaoyong Li[1], Xianhong Wei[1], Ying Cai[1], Bingshan Ma[1], Caiqin Hou[1], Xilian Han[1] and Liang Hong[2]

Abstract

Elementary siphons play an important role in designing deadlock prevention policies for flexible manufacturing systems modeling by Petri nets. This article proposes a deadlock control algorithm with maximally reachable number to cope with deadlock problems in ordinary Petri nets. First, we solve all elementary siphons and dependent siphons and then add both a control place and a control transition to each elementary siphon so that an extended net system (N', M') is obtained. Second, by constructing an integer programming problem of *P*-invariants of (N', M'), the controllability test for dependent siphons in N' is performed via this integer programming problem. Accordingly, a few of control places and control transitions are added for those dependent siphons that do not meet controllability as well. Therefore, a live controlled system (N^*, M^*) with maximally reachable number rather than number of maximally permissive behavior can be achieved. The correctness and efficiency of the proposed deadlock control algorithm is verified by a theoretical analysis and several examples that belong to ordinary Petri nets. Unlike these deadlock prevention policies with number of maximally permissive behavior in the existing literature, the proposed deadlock control algorithm can generally obtain a live controlled system (N^*, M^*) whose reachable number is the same as that of an original uncontrolled net (N_0, M_0), that is, maximally reachable number is greater than number of maximally permissive behavior.

Keywords

Flexible manufacturing system, Petri nets, deadlocks, elementary siphons, maximally reachable number, number of maximally permissive behavior

Date received: 21 February 2017; accepted: 14 August 2017

Handling Editor: ZhiWu Li

Introduction

A flexible manufacturing system (FMS) is a highly technical and synthetic system consisting of some intelligent subsystems, including an overall and reasonable configuration of some numerically controlled machines, a number of working units for material handling and products assembling, and a central computer workstation that can sample input information from sensors and then send output control signals to actuators

[1]School of Civil Engineering, Lanzhou University of Technology, Lanzhou, People's Republic of China
[2]College of Electronics and Information, Xi'an Polytechnic University, Xi'an, People's Republic of China

Corresponding author:
Shaoyong Li, School of Civil Engineering, Lanzhou University of Technology, No. 287, Langongping Road, Lanzhou 730050, Gansu, People's Republic of China.
Email: lishaoyong99@163.com

online. Because of the competition of shared manufacturing resources, a highly undesirable phenomena in a highly automated FMS, namely, deadlocks[1–6] occur definitely when some processing steps or jobs are trapped into an endless loop waiting for service which occupied by other processing ones, and vice versa, leading to the global or local block for an FMS. Therefore, it is necessary to consider the technical solutions to deadlock problems when designing an FMS.[7–13] In the past two decades, the research of resolving deadlocks has become the hot issues and many scholars and researchers have achieved a lot of remarkable results in theory and practical applications.[6,14–26]

Because of their own characteristics to easily and concisely describe the concurrent execution of processes and the reasonable distribution of shared resources in an FMS,[12,27] Petri nets have been widely used to model, analyze, and simulate the static and dynamic behaviors of an FMS, especially in deadlock problems. Using Petri nets, three policies, called deadlock detection and recovery (DDR),[1,28] deadlock avoidance (DA),[1,4] and deadlock prevention (DP)[6,10,23–25,28–32], respectively, are developed to cope with deadlock problems in FMSs. DDR can detect deadlocks and permit their occurrence and then adopts the corresponding measures to recover an FMS to a correct state. A DDR strategy is suitable for such case in which deadlocks are infrequent and their consequence is not serious. But it requires the time cost for detection and recovery. As for a DA technique, a shared resource is granted to a process only if the resulting state in an FMS is not a deadlock, which is an online computational mechanism to guarantee correct system evolutions. However, DP is an offline method that can consider and solve deadlocks in design and planning stages for an FMS. That is to say, its execution requires no run-time cost and guarantees the correct evolutions of different processes or jobs in an FMS, so a deadlock prevention policy (DPP) is widely used to design a liveness-enforcing supervisors to eliminate deadlocks in an FMS. Structural analysis and reachability graph (RG) are two major methods that belong to DPP.[1] In addition, as structural objects of a Petri net, siphons are closely related to its deadlock, deadlock-free, and liveness. Thus, DPP are classified into two categories: siphon-based method (SBM)[25,29,30,33–35] and reachability graph–based method (RGBM).[6,10,32,36–38] Generally, behavioral permissiveness (BP), structural complexity (SC), and computational complexity (CC) are three major criteria to evaluate SBM or RGBM. By means of explicit enumeration of all state nodes of an RG of the given Petri net model (PNM), RGBM can divide the corresponding RG into two parts: deadlock zone (DZ) including deadlocks and critical bad states that inevitably lead to deadlocks and a deadlock-free zone (DFZ) representing the good states that definitely keep the correct evolutions of PNM. As for a given PNM, these DPPs using RGBM[15,36,38] can generally find an optimal liveness-enforcing supervisor that ensures every state node within DFZ rather than DZ to be reached. That is to say, for a resultant controlled net system with liveness, its reachable states excluding deadlocks and bad states are called maximally permissive behavior (MPB), and the corresponding number of reachable states for this live controlled Petri net system is number of maximally permissive behavior (NMPB), which usually means the full use of the system resources. However, such RGBM usually suffers from a state explosion problem of the corresponding RG for a large-sized PNM with the large initial markings.[38] In order to cope with the state explosion problem, a few novel methods are successively developed to design the liveness-enforcing supervisors with MPB. Among them, Chen and Li[15] propose a non-iterative approach to design a maximally permissive control place (CP) by a place invariant (PI) that forbids one of the first-met bad markings (FBMs) and none of legal markings is forbidden. However, the computation for such PI needs solving an integer linear programming problem (ILPP), which may cause CC. Thus, in order to overcome the complexity of this method, a vector covering approach (VOA) is developed to reduce the sets of legal markings and FBM to be small, which are considered in the design of a liveness-enforcing supervisor. As a result, a maximally permissive liveness-enforcing supervisor within the class of supervisors where each CP is associated with a *P*-semiflow can be obtained if the ILPP has a solution. Accordingly, Chen et al.[13,36,37] also use a VOA to first compute a minimal covering set of legal markings and a minimal covered set of FBMs, and then these two minimal sets are considered for an iterative process of designing CPs that is different from Chen and Li.[15] Finally, a maximally permissive supervisor with a small number of CPs can be obtained when all FBMs that are forbidden by the PI are removed from the minimal covered set of FBMs, which obviously reduces the computational time compared with their previous work. Uzam and Li[6] adopt a set of mixed integer programming (MIP) formulations to perform an iterative extraction for bad markings in an RG of the uncontrolled PNM such that the explicit enumeration of all nodes of the RG is avoided in contrast with the iterative method of synthesizing liveness-enforcing supervisors,[38] leading to a controlled PNM with liveness, that is, MPB and has a relatively simple structure. By a lexicographic multiobjective integer programming problem, B Huang et al.[39] design a MPB (optimal) supervisor while forbidding all FBMs and permitting all legal markings in a Petri net model. In the meantime, a conversion method is proposed to convert the nonlinear model that is associated with minimizing its structure into a linear one. Hence, a liveness-enforcing supervisor with MPB and minimal structure is obtained. However, the corresponding computational loads are not reduced to a great degree, and a few reductant control places (RCPs) may exist in the live controlled system (N^*, M^*), that is, CC and SC for these novel RGBM are not observably improved except for BP.

However, many scholars focus on structural analysis, such as siphons[7,9,20,25,29] and resource transition circuits,[10,35] and have developed a large number of DPPs. Among these representative DPPs, E-policy[29] adopts the complete siphon enumeration method (CSEM) to solve all siphons causing deadlocks in an S^3PR (systems of simple sequential processes with resources) and then adds a CP to each emptied siphon to prevent itself from being emptied. Thus, E-policy is first regarded as obtaining a live controlled Petri net system by SBM, although its performance is not ideal in terms of an overall evaluation of CC, SC, and BP. In order to reduce the SC of a liveness-enforcing supervisor, Li and Zhou[22,23] pioneered the concept of elementary siphons (ESs) and dependent siphons (DSs). Li and Zhou-policy only adds the CPs to all ESs and some DSs that cannot meet the controllability and then achieves a liveness-enforcing supervisor, whose SC is enormously depressed compared with that of E-policy. Subsequently, Li and Zhou[9] and Chao and Chen[14] and Chao[40] propose the further research results including the novel methods to compute ESs, controllability of DSs (or compound siphons), and the liveness conditions associated with the controlled ESs and DSs for S^3PR and S^3PGR (systems of simple sequential processes with general resources requirement), which enrich the theory of elementary siphon and its applications. But BP and CC are not ideal for the liveness-enforcing supervisors synthesized from the controlled ESs and DSs. In the meantime, by MIP[3,8] or the revised MIP[7] (Li and Li 2012c), the partial siphon enumeration method (PSEM) is also used to solve those siphons that definitely cause deadlocks so that the computational loads for the related DPPs such as C-policy,[7] H-policy,[20,31] PR-policy,[4] P-policy,[25] and LL-policy,[28,32,33,41] obviously decrease. Moreover, several novel methods of controlling siphons, for example, $\max, \max', \max''$, and $\max^*$-controllability for siphons[12] are successively proposed and employed for these solved siphons in order to design the corresponding DPPs; these methods gradually improve the behavioral permissiveness of the finally live controlled Petri net system to a great degree. In short, the liveness-enforcing supervisors designed by PSEM can yield an overall improvement in CC, BP, and SC than those ones constructed by CSEM. Especially, it is the first time that Chao[40] add a CP and a control transition (CT) to each solved siphon in an S^3PR system with deadlocks, resulting in a live controlled Petri net system with maximally reachable number (MRN). That is to say, reachable state number of the final controlled PNM with liveness is the same as that of the uncontrolled one with deadlocks, that is, MRN > NMPB, and its BP is improved to an ideal degree by such method of adding both CPs and CTs to these siphons causing deadlocks.[40] However, the difference of controllability for ESs and DSs is not considered in Chao,[40] which may lead to a live controlled PNM (N^*, M^*) with a relatively complicated structure.

Partially enlightened by the advantages of using ESs[14,22,23] and the addition of CPs and CTs,[40] the purpose of this work is to design a deadlock control algorithm (DCA) to obtain not only a liveness-enforcing supervisor with MRN but also a relatively simple structure in order to eliminate deadlocks in the typical classes of ordinary Petri nets (OPNs), namely, S^3PR,[29] linear S^3PR $(L-S^3PR)$,[31] and extended S^3PR (ES^3PR)[8,31] comparing with the method in Chao.[40] The proposed DCA focuses on solving ESs and DSs, executing controllability test for all DSs, and adding CPs and CTs for all ESs and a few of DSs properly, which is executed in two stages. At the first stage, all ESs and DSs in an uncontrolled system that belong to OPNs are solved and classified by the methods in Chao and Chen[14] and Li and Zhou.[22] By adding a CP and CT[40] to each ESs, all ESs are controlled by the corresponding CPs and CTs so that an extended net system (N', M') is obtained. Second, by constructing an integer programming problem (IPP) of P-invariants of (N', M'), a controllability test for all DSs in N' is performed via this IPP. If all DSs meet the desired controllability condition, then a live controlled system (N^*, M^*) is achieved directly, implying that the extended net system (N', M') is live. Conversely, the corresponding CPs and CTs are added for those DSs that cannot meet the controllability. Therefore, a live controlled system (N', M') can be achieved as well. A theoretical analysis and several examples belonging to S^3PR, linear S^3PR (L-S^3PR), and ES^3PR in the existing literature are used to illustrate the correctness and efficiency of the proposed DCA. Unlike these DPPs with NMPB in the existing literature, the proposed DCA can generally obtain a live controlled system (N', M') and its reachable state number is the same as that of the original uncontrolled net (N_0, M_0), that is, MRN is greater than NMPB. In addition, only adding the corresponding CPs and CTs for all ESs and a few of DSs that cannot pass through the controllability test under the proposed IPP implies that SC of the final live controlled system (N^*, M^*) can be reduced to some degree than that of the corresponding (N^*, M^*) in the study of Chao.[40]

The remainder of this article is organized as follows. The following section, "Preliminaries," presents the necessary background on Petri nets and the definitions of ES, deadlock states (DS), S^{3}PR, and so on. The method of adding CPs and CTs (resp. CP and CT)[40] is first reviewed in section "A two-stage deadlock control policy." Accordingly, an IPP to test the controllability for all DSs in N' and a two-stage DCA with MRN are developed in the same section. Section "Examples" illustrates efficiency of the proposed DCA through several examples belonging to OPNs. The last section "Conclusion" concludes this article.

Preliminaries

Some relevant basic concepts of Petri nets

A Petri net is a four-tuple $N = (P, T, F, W)$, where P and T are finite and nonempty sets, respectively. P is a set of places and T is a set of transitions with $P \cup T \neq \emptyset$ and $P \cap T = \emptyset$. $F \subseteq (P \times T) \cup (T \times P)$ is called the flow relation or the set of directed arcs. $W : (P \times T) \cup (T \times P) \rightarrow \mathbb{N}$ is a mapping that assigns a weight to an arc: $W(x,y) > 0$ if $(x,y) \in F$, and $W(x,y) = 0$ otherwise, where $x, y \in P \cup T$ and $\mathbb{N}$ denote the set of non-negative integers. $N = (P, T, F, W)$ is called an ordinary net, denoted as $N = (P, T, F)$, if $\forall f \in F, W(f) = 1$. $N = (P, T, F, W)$ is called a generalized net, if $\exists f \in F, W(f) > 1$. Given a node $x \in P \cup T$, ${}^{\bullet}x = \{y \in P \cup T | (y,x) \in F\}$ is called the preset of x, while $x^{\bullet} = \{y \in P \cup T | (x,y) \in F\}$ is called the postset of x. A marking is a mapping $M : P \rightarrow \mathbb{N}$. (N, M_0) is called a marked Petri net or a net system. The set of markings reachable from M in N is denoted as $R(N, M)$. Incidence matrix $[N]$ of net N is a $|P| \times |T|$ integer matrix with $[N](p,t) = W(t,p) - W(p,t)$, where $|P|$ and $|T|$ denote the number of places and transitions in N, respectively. Given a Petri net (N, M_0), $t \in T$ is live under M_0 if $\forall M \in R(N, M_0), \exists M' \in (N, M), M'[t\rangle$. (N, M_0) is live if $\forall t \in T, t$ is live under M_0. (N, M_0) is dead under M_0 if $\nexists t \in T, M_0[t\rangle$. (N, M_0) is deadlock-free (weakly live) if $\forall M \in R(N, M_0)$, $\exists t \in T, M[t\rangle$.[1]

A nonempty set $S \subseteq P$ is a siphon (trap) if ${}^{\bullet}S \subseteq S^{\bullet}$ ($S^{\bullet} \subseteq {}^{\bullet}S$). A siphon is minimal if there is no siphon contained in it as a proper subset. A minimal siphon is called a strict minimal siphon (SMS) if it does not contain a trap. Π is used to denote the set of SMS in a Petri net. A $P(T)$-vector is a column vector $I(J) : P(T) \rightarrow Z$ indexed by $P(T)$, where Z is the set of integers. I is a P-invariant (called P-inv for short) if $I \neq \mathbf{0}$ and $I^T[N] = \mathbf{0}^T$. P-inv I is said to be a P-semiflow if every element of I is non-negative. $\|I\| = \{p \in P | I(p) \neq 0\}$ is called the support of I. $\|I\|^+ = \{p | I(p) > 0\}$ denotes the positive support of I. $\|I\|^- = \{p | I(p) < 0\}$ denotes the negative support of I. Siphon S is inv-controlled by P-inv I under M_0 if $I^T M_0 > 0$ and $\{p \in P | I(p) > 0\} \subseteq S$. For economy of space, $\sum_{p \in P} M(p)p$, $\sum_{p \in \|I\|} I(p)p$, and $\sum_{t \in \|J\|} J(t)t$ denote a marking M, P-vector I, and T-vector J, respectively.[12] For example, $M = (12000)^T$ is written as $M = p_1 + 2p_2$.

Elementary siphon and dependent siphon

Definition 1. Let $S \subseteq P$ be a subset of places of Petri net $N = (P, T, F, W)$. P-vector λ_S is called the characteristic P-vector of S if $\forall p \in S$, $\lambda_S(p) = 1$; otherwise $\lambda_S(p) = 0$.[22]

Definition 2. Let $S \subseteq P$ be a subset of places of Petri net $N = (P, T, F, W)$ and λ_S be the characteristic P-vector of S.[22] T-vector η_S is called the characteristic T-vector of S if $\eta_S = [N]^T \lambda_S$, where $[N]^T$ is the transpose of incidence matrix $[N]$.

Definition 3. Let $N = (P, T, F, W)$ be a net with $|P| = m$, $|T| = n$, and k siphons, $S_1, S_2, \ldots$ and $S_k, m, n, k \in \mathbb{N}^+ = \{1, 2, \ldots\}$.[22] Let $\lambda_{S_i}(\eta_{S_i})$ be the characteristic $P(T)$-vector of siphon $S_i, i \in \mathbb{N}_k = \{1, 2, \ldots, k\}$. We define $[\lambda]_{k \times m} = [\lambda_{S_1} | \lambda_{S_2} | \cdots | \lambda_{S_k}]^T$ and $[\eta]_{k \times n} = [\lambda]_{k \times m} \times [N]_{m \times n} = [\eta_{S_1} | \eta_{S_2} | \cdots | \eta_{S_k}]^T$. $[\lambda]$ ($[\eta]$) is called the characteristic $P(T)$-vector matrix of the siphons in N.

Definition 4. Let $\eta_{S_\alpha}, \eta_{S_\beta}, \ldots$ and $\eta_{S_\gamma}(\{\alpha, \beta, \ldots, \gamma\} \subseteq \mathbb{N}_k)$ be a linearly independent maximal set of matrix $[\eta]$. Then, $\Pi_E = \{S_\alpha, S_\beta, \ldots, S_\gamma\}$ is called a set of ESs in N.[22]

Definition 5. $S \notin \Pi_E$ is called a strongly dependent siphon if $\eta_S = \Sigma_{S_i \in \Pi_E} a_i \eta_{S_i}$, where $a_i \geq 0$.[22]

Definition 6. $S \notin \Pi_E$ is called a weakly dependent siphon if $\exists A, B \subset \Pi_E$ such that $A \neq \emptyset, B \neq \emptyset, A \cap B = \emptyset$ and $\eta_S = \Sigma_{S_i \in A} a_i \eta_{S_i} - \Sigma_{S_j \in B} b_j \eta_{S_j}$, where $a_i, b_j > 0$.

The above method proposed by Li and Zhou needs to find all SMS, and then ESs and DSs are determined from Definitions 3–6. Although the number of SMS grows exponentially with respect to the size of a given PNM, all SMS in a PNM are conveniently found using Integrated Net Analyzer (INA), a popular Petri net analysis tool.[42] An S^{3}PR is used to model a large class of FMS. Its definition is briefly given as follows[29]

Definition 7. An S^{3}PR is defined as the union of a set of nets $N_i = (P_{A_i} \cup \{p_i^0\} \cup P_{R_i}, T_i, F_i)$, $i \in \{1, 2, \ldots, m\}$, sharing common places, where the following statements are true:

1. p_i^0 is called the process idle place of N_i. $p \in P_{A_i}$ and $r \in P_{R_i}$ are called operation and resource

place, respectively, where P_{A_i} is a set of operation places and P_{R_i} is a set of resource places.

2. $P_{R_i} \neq \emptyset, P_{A_i} \neq \emptyset, p_i^0 \notin P_{A_i}, (P_{A_i} \cup \{p_i^0\}) \cap P_{R_i} = \emptyset$.
   (a) $\forall p \in P_{A_i}, \forall t \in {}^\bullet p, \forall t' \in p^\bullet, \exists r_p \in P_{R_i}, {}^\bullet t \cap P_{R_i} = t'^\bullet \cap P_{R_i} = \{r_p\}$;
   (b) $\forall r \in P_{R_i}, {}^{\bullet\bullet} r \cap P_{A_i} = r^{\bullet\bullet} \cap P_{A_i} \neq \emptyset$ and $\forall r \in P_R, {}^\bullet r \cap r^\bullet = \emptyset$;
   (c) ${}^{\bullet\bullet}(p_i^0) \cap P_{R_i} = (p_i^0)^{\bullet\bullet} \cap P_{R_i} = \emptyset$.
3. N_i' is a strongly connected machine, where $N_i' = (P_{A_i} \cup \{p_i^0\} \cup P_{R_i}, T_i, F_i)$ is the resultant net after the places in P_{R_i} and related arcs are removed from N_i.
4. Every circuit of N contains the place p_i^0.
5. Any two nets N_1 and N_2 are composable, denoted as $N_1 \circ N_2$, if they share a set of common resource places. Every shared place must be resource one.
6. Transitions in $(p_i^0)^\bullet$ and ${}^\bullet(p_i^0)$ are called the source and sink transitions of an S^3PR, respectively.

Definition 8. Let S be a siphon in a marked S^3PR with $S = S_R \cup S_A$, $S_R = S \cap P_{R_i}$, and $S_A = S \backslash S_R$, where S_A and S_R are called the set of operation and resource places for S, respectively. For $r \in P_{R_i}$, $H(r) = {}^{\bullet\bullet} r \cap P_{A_i}$, the operation places that use r is called the set of holders of r. $[S] = (\cup_{r \in S_R} H(r)) \backslash S$ is called the complementary set of S.

Moreover, as for the standard definitions of $L - S^3PR$ and ES^3PR, the reader is referred to the literature.[8,22,31]

For example, we can find three SMS in the net shown in Figure 1 by INA[42] where $S_1 = \{p_3, p_6, p_9, p_{10}\}$, $S_2 = \{p_4, p_7, p_{10}, p_{11}\}$, and $S_3 = \{p_4, p_6, p_9, p_{10}, p_{11}\}$. From Definitions 1 and 2, $\lambda_{S_1} = (0,0,1,0,0,1,0,0,1,1,0)^T$, $\lambda_{S_2} = (0,0,0,1,0,0,1,0,0,1,1)^T$, $\lambda_{S_3} = (0,0,0,1,0,1,0,0,1,1,1)^T$, $\eta_{S_1} = (-1,1,0,0,0,1,-1,0)^T$, $\eta_{S_2} = (0,-1,1,0,0,0,1,-1)^T$, and $\eta_{S_3} = (-1,0,1,0,0,1,0,-1)^T$ are obtained, respectively. It can be seen that $\eta_{S_3} = \eta_{S_1} + \eta_{S_2}$. Hence, S_1 and S_2 are two ESs, and S_3 is a strongly DS due to Definitions 3–5. In addition, $[S_1] = \{p_2, p_7\}$, $[S_2] = \{p_3, p_8\}$, and $[S_3] = \{p_2, p_3, p_7, p_8\}$ are obtained from Definition 8.

Theorem 1. An S^3PR (N, M_0) is live iff $\forall M \in R(N, M_0), \forall S \in \Pi, M(S) > 0$.[1,12]

Theorem 2. An ES^3PR (N, M_0) is live iff no siphon in it can become empty.[12,31]

Theorem 3. Let $L = \{\sigma | M_0[\sigma > M', M' \in R(N_0, M_0), M'(S) > 0, \forall S \in \Pi\}$ be the language of all firing sequences in (N_0, M_0) with no siphons ever empty, $L^* = \{\sigma | M_0^*[\sigma > M'^*, M'^* \in R(N^*, M^*), M'^*(S) > 0, \forall S \in \Pi\}$ be the language of all firing sequences in (N^*, M^*) with no siphons ever empty, and $R^{|N}(N^*, M_0^*) = \{M^{*|N} | M^* \in R(N^*, M^*)\}$ be the projection of $R(N^*, M^*)$ onto $R(N_0, M_0)$, where $M^{*|N}$ is the projection of M^* onto M, that is, $\forall p \in P, M^{*|N}(p) = M^*(p)$, and $M^{*|N}$ is a $||P||$-dimensional vector, and P is the set of places in N_0.[40] Then, $L = L^*$ and $R^{|N}(N^*, M_0^*) = R(N_0, M_0)$ (i.e. the controlled model reaches the same number of states as the uncontrolled model).

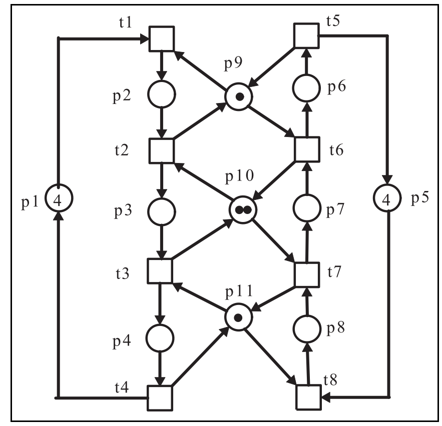

Figure 1. A non-live marked S^3PR.

A two-stage deadlock control policy

The method of adding CP and transition

As stated in Introduction, Chao[40] proposes a novel idea to add a CP and CT but not a CP to each solved SMS in an S^3PR system with deadlocks, resulting in a live controlled Petri net system with MRN. For each $S_i \in \Pi$ and the corresponding $[S_i]$, the method of adding corresponding CP (denoted as V_{S_i}) with $M_0(V_{S_i}) = 0$ and CT (denoted as t_{S_i}) for S_i is briefly described as follows:

1. Connection between t and V_{S_i}. (a) For each $t \in {}^\bullet[S_i] \backslash [S_i]^\bullet$, there exists an arc (t, V_{S_i}) with $W(t, V_{S_i}) = 1$ from t to V_{S_i}; (b) for each $t \in [S_i]^\bullet \backslash {}^\bullet[S_i]$, there exists an arc (V_{S_i}, t) with $W(V_{S_i}, t) = 1$ from V_{S_i} to t.
2. For V_{S_i} and t_{S_i}, there exists an arc (t_{S_i}, V_{S_i}) with $W(t_{S_i}, V_{S_i}) = M_0(S_i) - 1$ from t_{S_i} to V_{S_i} and an arc (V_{S_i}, t_{S_i}) with $W(V_{S_i}, t_{S_i}) = M_0(S_i)$ from V_{S_i} to t_{S_i}, respectively.
3. Connection between t_{S_i} and the related places in the production routes PR_j, where j represents the sequence number for PR, $j = 1, 2, \ldots, m$. (a) For each $p_{S_i} \in {}^\bullet(V_{S_i}{}^\bullet) \cap PR_j$, there exists an arc (p_{S_i}, t_{S_i}) with $W(p_{S_i}, t_{S_i}) = 1$ from p_{S_i} to t_{S_i}; (b) for each $r_{S_i} \in P_{R_i}$ and $p_{S_i} \in H(r_{S_i})$, there exists an arc (t_{S_i}, r_{S_i}) with $W(t_{S_i}, r_{S_i}) = 1$ from t_{S_i} to r_{S_i}; (c)

for each $p_j^0 \in \mathrm{PR}_j^0$, there exists an arc (t_{S_i}, p_j^0) with $W(t_{S_i}, p_j^0) = 1$ from t_{S_i} to p_j^0. In essence, PR_j is the same as N_i in Definition 7.

4. For each other $S_k \in \Pi$ and $S_k \neq S_i$, there exists an arc (V_{S_k}, t_{S_i}) with $W(V_{S_k}, t_{S_i}) = 1$ from V_{S_k} to t_{S_i} when $p_{S_i} \in [S_k]$ and $p_{S_i} \in {}^\bullet(V_{S_i}{}^\bullet) \cap \mathrm{PR}_j$.

Note that as for an S_i, its corresponding p_{S_i} (r_{S_i}) is generally not unique due to different PR_j, that is, $|p_{S_i}|(|r_{S_i}|) \geq 2$. For example, there are two production routes PR_1 and PR_2 in the net shown in Figure 1. For S_1, corresponding to PR_1 and PR_2, we have $p_{S_1} = \{p_2\}$ $(r_{S_1} = \{p_9\}, p_j^0 = \{p_1\})$ and $p_{S_1} = \{p_7\}$ $(r_{S_1} = \{p_{10}\}, p_j^0 = \{p_5\})$ in terms of the third rule mentioned above, respectively. Similarity, for S_2, corresponding to PR_1 and PR_2, $p_{S_2} = \{p_3\}$ $(r_{S_2} = \{p_{10}\}, p_j^0 = \{p_1\})$ and $p_{S_2} = \{p_8\}$ $(r_{S_2} = \{p_{11}\}, p_j^0 = \{p_5\})$ are obtained, respectively. Also, for S_3, corresponding to PR_1 and PR_2, $p_{S_3} = \{p_3\}$ $(r_{S_3} = \{p_{10}\}, p_j^0 = \{p_1\})$ and $p_{S_3} = \{p_7\}$ $(r_{S_3} = \{p_{10}\}, p_j^0 = \{p_5\})$, respectively. But how to select the corresponding p_{S_i} (r_{S_i}) for an S_i is not clear in the study of Chao.[40]

Since the number of PR_j is obviously less than that of the solved SMS, this article adopts a simple method, called circulating sequence number (CSN) of the production routes PR_j, to assign the corresponding PR_j to an S_i $(j = 1, 2, \ldots, m,\ i = 1, 2, \ldots, n, m < n)$ in turn. That is to say, $\mathrm{PR}_1, \mathrm{PR}_2, \ldots, \mathrm{PR}_m$, PR_1, ... and PR_l corresponds $S_1, S_2, \ldots, S_m, S_{m+1}, \ldots$, and PR_n, respectively, where $m + l = n$. Reconsidering an example depicted in Figure 1, PR_1, PR_2, and PR_1 are assigned to the corresponding S_1, S_2, and S_3 by CSN. Accordingly, $p_{S_1} = \{p_2\}$ $(r_{S_1} = \{p_9\}, p_j^0 = \{p_1\})$ for S_1, $p_{S_2} = \{p_8\}$ $(r_{S_2} = \{p_{11}\}, p_j^0 = \{p_5\})$ for S_2, and $p_{S_3} = \{p_3\}$ $(r_{S_3} = \{p_{10}\}, p_j^0 = \{p_1\})$ for S_3 are determinated clearly. As a result, by means of the above methods of adding CPs and CTs and CSN, a controlled Petri net system is obtained, which is shown in Figure 2.

The controlled Petri net system shown in Figure 2 is verified to be live by INA.[42] Its MRN is 47*, the same as the reachable number of original marked S^3PR with deadlocks and is greater than the corresponding NMPB (42).

An IPP to test controllability of dependent siphons

From the above mentioned, adding a CP and CT to each solved SMS[40] can obtain a live controlled Petri net system (N^*, M^*) with MRN. On the basis of Definitions 1–6, SMSs are classified into ESs and DSs. From the existing literature, we know that those policies to make ESs explicitly controlled and DSs implicitly controlled, respectively, can reduce the SC of finally live controlled Petri net systems to some degree. However, this method of adding a CP and CT to each solved SMS[40] without considering different controllability between ESs and DSs may result in a relatively complex structural (N^*, M^*) with MRN.

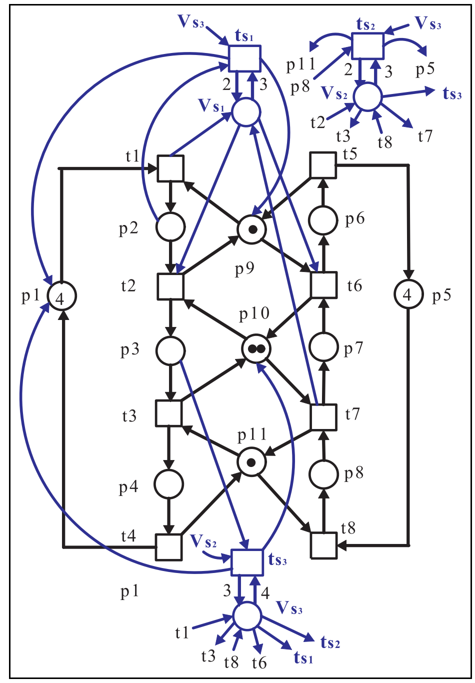

Figure 2. A controlled Petri net system.

Partially motivated by the advantages of ESs to design deadlock control policies,[14,22–24] an IPP of P-invariants of an extended net system (N', M') is constructed to test controllability of all DSs in (N', M') formed by all ESs which are controlled by the addition of CPs and CTs. By this IPP test, a few CPs and CTs are added to those0 DSs that cannot meet controllability, and the remaining DSs meeting controllability are implicitly controlled. In other words, the implicitly controlled DSs do not need the addition of CPs and CTs. Thus, the SC of a finally live controlled Petri net system (N^*, M^*) can be reduced to some degree.

Assume that $\Pi_{ES} = \{ES_1, ES_2, \ldots, ES_k\}$ and $\Pi_{DS} = \{DS_1, DS_2, \ldots, DS_m\}$ are a set of ESs and a set of DSs in an original S^3PR (N_0, M_0), respectively. After all ESs are controlled by the addition of CPs and CTs, an extended net system (N', M') and its P-invariants $I_i (i = 1, 2, \ldots, l)$ can be obtained. As for each DS_j, we propose its controllability test in this article, which focuses on $\sum_{p \in DS_j} M'(p) > \sum_{p \in [DS_j]} M'(p), M' \in R(N_0, M_0)$, where $\sum_{p \in DS_j} M'(p)$ is the number of tokens held by DS_j at M', and $\sum_{p \in [DS_j]} M'(p)$ is the number of tokens required by $[DS_j]$ at the same reachable marking. Since the places in $[DS_j]$ compete for tokens with the operation places in DS_j, the total number of tokens held by DS_j and $[DS_j]$ keeps constant for any reachable marking. It is inferred that DS_j is marked at $M' \in R(N_0, M_0)$ if $\sum_{p \in DS_j} M'(p) > \sum_{p \in [DS_j]} M'(p)$, where $\sum_{p \in DS_j} M'(p)$

(denoted as $M'(DS_j)$) and $\sum_{p\in[DS_j]} M'(p)$ (denoted as $M'([DS_j])$) are solved by the following integer programming

$$\max\left(\sum_{p\in DS_j} M'(p) + \sum_{p\in[DS_j]} M'(p)\right) \quad (1)$$

subject to

$$I_i^T \bullet M' = I_i^T \bullet M_0, i = 1, 2, \ldots, l \quad (2)$$

$$M' = M_0 + [N]Y, M' \geq 0, Y \geq 0 \quad (3)$$

I_i is the minimal P-invariants of N'. Maximizing $\sum_{p\in[DS_j]} M'(p)$ shows the maximum ability of $[DS_j]$ to occupy tokens from resource places of DS_j. Similarly, Maximizing $\sum_{p\in DS_j} M'(p)$ implies that DS_j can prevent tokens to flow into $[DS_j]$ to the most degree. Equation (2) shows that any one of P-invariants of N' corresponds to a set of places whose weighted token count is a constant for any reachable marking. The interrelation between the marking's changes and transition occurrences follows from equation (3), the state equation of a Petri net (N, M).

Proposition 1. Let (N_0, M_0) be an uncontrollable marked S^3PR, $ES_1, ES_2, \ldots, ES_k$ be the ESs of N_0, and $DS_1, DS_2, \ldots, DS_m$ be the dependent siphons of N_0. An extended net system (N', M') is obtained by adding k CPs and CTs for k ESs, where $\Theta = \{I_i | i = 1, 2, \ldots, l\}$ is the set of minimal P-invariants of N'. If $M'(DS_j) > M'([DS_j]), M' \in R(N_0, M_0), j = 1, 2, \ldots, m$, then DS_j can be implicitly controlled in N'.

Proof. Since the set of SMS Π in an original S^3PR (N_0, M_0) can be computed by means of INA;[42] according to Definitions 1–6, $\Pi_{ES} = \{ES_1, ES_2, \ldots, ES_k\}$ and $\Pi_{DS} = \{DS_1, DS_2, \ldots, DS_m\}$ can be obtained from Π. By the method stated in the subsection "The method of adding CP and transition," the addition of k CPs and CTs makes k ESs explicitly controlled and leads to an extended net system (N', M'). For each DS_j of N_0, DS_j is also a dependent siphon of N' and keeps the same complementary set $[DS_j]$. Although the places in $[DS_j]$ compete for tokens with the operation places in DS_j, the total number of tokens held by DS_j and $[DS_j]$ keeps constant at any reachable marking M. A controllability test for Π_{DS} is performed on the basis of equations (1)–(3). When $M'(DS_j) > M'([DS_j]), M' \in R(N_0, M_0)$, it is obvious that the number of tokens held by DS_j at M' is greater than the required number of tokens for $[DS_j]$ at M', indicating that DS_j can never be emptied at $M' \in R(N_0, M_0)$, that is, $M'(DS_j) > 0$. As a result, such DSs cannot affect the liveness of an extended net system (N', M') due to Theorems 1 and 2, implying that such DSs can be implicitly controlled in (N') without addition of CPs and CTs.

Although this IPP to test controllability of dependent siphons is time-consuming in theory, it is an established method and is performed offline. In addition, even for the large-sized models of OPNs, this IPP to test controllability of DSs is conveniently executed by LinGo[43] in less time.

A DCA using the directly controlled ESs and the identified and controlled DSs

On the basis of the above discussion and results, in order to eliminate deadlocks in an S^3PR, $L-S^3PR$, and ES^3PR that are typical representatives of OPNs and achieve a live controlled Petri net (N^*, M^*) with MRN, a DCA using the directly controlled ESs and the identified and controlled DSs is developed in this section and described below.

Algorithm 1. A DCA using the directly controlled ESs and the identified and controlled DSs.

Input: an uncontrollable marked S^3PR (N_0, M_0), $N_0 = (P_A \cup P_R \cup P^0, T, F, W)$.
Output: a live controlled Petri net (N^*, M^*) with MRN.
Step 1: By INA, compute all SMS S in an original S^3PR (N_0, M_0) and obtain $\Pi = \{S_1, S_2, \ldots, S_n\}$.
Step 2: According to Definitions 1–6, find $\Pi_{ES} = \{ES_1, ES_2, \ldots, ES_k\}$ and $\Pi_{DS} = \{DS_1, DS_2, \ldots, DS_m\}$, where $k + m = n$.
Step 3: Solve $\Pi_{[ES]} = \{[ES_1], [ES_2], \ldots, [ES_k]\}$ and $\Pi_{[DS]} = \{[DS_1], [DS_2], \ldots, [DS_m]\}$ in terms of Definition 8.
Step 4: As stated in Subsection "The method of adding CP and transition," add a CP and CT to each ES and an extended net system (N', M') is achieved.
Step 5: $i := 1$.
Step 6: **while** $i \leq |DS|$ **do** $/^*|DS|$ denotes the number of DSs $^*/$
Step 7: Based on Proposition 1 and LinGo, an IPP related to $M'(DS_i) > M'([DS_j]), M' \in R(N_0, M_0)$ is performed among DSs.
Step 8: $i := i + 1$.
Step 9: end while
Step 10: **if** $M'(DS) > M'([DS_j])$ for each DS **then**
Step 11: $N^* := N'$, $M^* := M'$ and go to Step 15.
Step 12: **else**
Step 13: As for those $DS_j (j = 1, 2, \ldots, q, q < |DS|)$ with $M'(DS_j) < M'([DS_j])$, $M' \in R(N_0, M_0)$, by the above method of addition of CPs and CTs, add the corresponding CPs and CTs to them in (N', M').
Step 14: **end if**
Step 15: Output (N^*, M^*).

Algorithm 1 is performed in two stages and can be briefly explained as follows. At the first stage, all SMS causing deadlocks in an original S^3PR (N_0, M_0) are computed by INA and then $\Pi = \{S_1, S_2, \ldots, S_n\}$ is obtained. $\Pi_{ES} = \{ES_1, ES_2, \ldots, ES_k\}$ and $\Pi_{DS} = \{DS_1, DS_2, \ldots, DS_m\}$ are solved, respectively, on the basis of Definitions 1–6. Also, from Definition 8, $\Pi_{[ES]} = \{[ES_1], [ES_2], \ldots, [ES_k]\}$ and $\Pi_{[DS]} = \{[DS_1], [DS_2], \ldots, [DS_m]\}$ are obtained. Accordingly, an extended net system (N', M') is achieved by adding a CP and CT to each ES. At the second stage, an IPP is performed to test whether each DS meet $M'(DS) > M'([DS_j]), M' \in R(N_0, M_0)$. If all DSs meet such controllability, then all DSs are implicitly controlled without addition of CPs and CTs, implying that an extended net system (N', M') is live. In a word, (N', M') is (N^*, M^*) and Algorithm 1 terminates the execution. Otherwise, the corresponding CPs and CTs are added to those DSs that cannot meet such controllability so that an extended net system (N', M') is further transformed into a live controlled net system (N^*, M^*).

Proposition 2. Let (N_0, M_0) be a marked S^3PR with deadlocks. Algorithm 1 is applied to it, which can output a live controlled net system (N^*, M^*) with MRN.

Proof. First, SMSs causing deadlocks can be found by INA[42] and then $\Pi_{ES} = \{ES_1, ES_2, \ldots, ES_k\}$, $\Pi_{DS} = \{DS_1, DS_2, \ldots, DS_m\}$, $\Pi_{[ES]} = \{[ES_1], [ES_2], \ldots, [ES_k]\}$, and $\Pi_{[DS]} = \{[DS_1], [DS_2], \ldots, [DS_m]\}$ are obtained, respectively, in terms of Definitions 1–6 and 8. By the above method of addition of CPs and CTs, each ES is explicitly controlled by adding a CP and CT, so Algorithm 1 outputs an extended net system (N', M'). Second, an IPP to test controllability of DSs is executed by LinGo.[43] For those DSs that cannot meet $M'(DS) > M'([DS_j]), M' \in R(N_0, M_0)$, the corresponding CPs and CTs are added to them due to Proposition 1 and then (N', M') is further transformed into a controlled net system (N^*, M^*). Because of the addition of CPs and CTs to all ESs and those DSs that cannot meet controllability, there is no emptied siphons in (N^*, M^*). That is to say, all solved ESs and DSs are sufficiently marked at $M^* \in R(N_0, M_0)$ via two stages of the addition of CPs and CTs. Finally, a controlled net system (N^*, M^*) is live due to Theorems 1 and 2. In addition, from Theorem 3, L and L^* are the language of all firing sequences in (N_0, M_0) and (N^*, M^*) with no siphons ever empty, respectively, $R^{|N}(N^*, M_0^*) = \{M^{*|N} | M^* \in R(N^*, M^*)\}$ is the projection of $R(N^*, M^*)$ onto $R(N_0, M_0)$. Algorithm 1 is performed in two stages of the addition of CPs and CTs, which leads to a fact that no emptied siphons exist in (N^*, M^*) and all reachable states for (N^*, M^*) can appear in $R^{|N}(N^*, M^*)$. So $L = L^*$ and $R^{|N}(N^*, M_0^*) = R(N_0, M_0)$ can be obtained due to Theorem 3. In another word, the reachable number of the live controlled system (N^*, M^*) is the same as that of an original uncontrolled net (N_0, M_0), implying that the truth of Proposition 2.

Since the number of SMSs in a net grows quickly and in the worst case grows exponentially with the size of the net[1,12,22] and the first stage of Algorithm 1 mainly concerns on solution and classification of SMSs, the complexity of Algorithm 1 is nondeterministic polynomial time (NP)-complete in theory. However, the related IPP to test controllability of DSs is conveniently solved by LinGo[43] during its second stage, and Algorithm 1 is totally performed offline. Therefore, on the basis of an overall analysis in theory and the practical application, Algorithm 1 is potential to design a DCA with MRN and a relatively simple structure; its efficiency can be shown via examples in the following section.

Examples

This section introduces some examples that belong to S^3PR and ES^3PR in the existing literature[20,22,25,29,41] to further exemplify Algorithm 1. Its control performance comparison with the existing approaches presented in the study of Ezpeleta et al.[29] (denoted as ECM),[18] (denoted as HHJ^+),[3,44] (denoted as HJX^+),[22,23] (denoted as LZ),[38] (denoted as UZ),[25] (denoted as PCF),[7] (denoted as $C1$),[40] (denoted as $C2$) and Li et al.[41] (denoted as LAW^+) is also carried out via some tables, where $*$, ⋇, R, DS, and DFS denote maximally reachable state, MPB, the ratio of the number of reachable states of the live controlled system (N^*, M^*) to MRN, deadlock states, and deadlock-free states of the corresponding Petri net system, respectively.

Let us reconsider an original S^3PR with deadlocks shown in Figure 1. Algorithm 1 is applied to it. Tables 1–6 show solution of ESs, DSs, and their complementary sets, the addition of CPs and CTs to ESs, controllability test to DSs, the supplemental addition of CPs and CTs to ESs and the addition of CPs and CTs to DSs, a two-stage deadlock control process using Algorithm 1, and its control performance comparison with the existing approaches, respectively.

Figure 3 shows a non-live marked ES^3PR, where $P^0 = \{p_1, p_8\}$, $P_A = \{p_2, p_3, p_4, p_5, p_6, p_7, p_9, p_{10}, p_{11}\}$, $P_R = \{p_{12}, p_{13}, p_{14}, p_{15}, p_{16}\}$.[31,41] We have $|P| = |P^0| + |P_A| + |P_R| = 16$, $|T| = 12$, and two production routes PR_m $(m = 1, 2)$, respectively.

Algorithm 1 is also applied to it. Tables 7–11 show solution of ESs, DSs, and their complementary sets, the addition of CPs and CTs to ESs, controllability test

Table 1. Two elementary siphons and one dependent siphon, their complementary sets, and presets and postsets of the complementary sets.

i	ES_i	$[ES_i]$	$^\bullet[ES_i]$	$[ES_i]^\bullet$
j	DS_j	$[DS_j]$	$^\bullet[DS_j]$	$[DS_j]^\bullet$
1	$\{p_3, p_6, p_9, p_{10}\}$	$\{p_2, p_7\}$	$\{t_1, t_7\}$	$\{t_2, t_6\}$
2	$\{p_4, p_7, p_{10}, p_{11}\}$	$\{p_3, p_8\}$	$\{t_2, t_8\}$	$\{t_3, t_7\}$
1	$\{p_4, p_6, p_9, p_{10}, p_{11}\}$	$\{p_2, p_3, p_7, p_8\}$	$\{t_1, t_8\}$	$\{t_3, t_6\}$

Table 2. The addition of CPs and CTs to two elementary siphons.

ES_i	$p_{ES_i} \in {}^\bullet(V_{ES_i}{}^\bullet) \cap PR_m$	p_1^0, p_2^0	r_{ES_i}	$^\bullet t_{ES_i}$	$t_{ES_i}{}^\bullet$	$^\bullet V_{ES_i}$	$V_{ES_i}{}^\bullet$
ES_1	$\{p_2\}\surd, \{p_7\}$ PR$_1\surd$, PR$_2$	$\{p_1\}\surd, \{p_5\}$	$\{p_9\}\surd, \{p_{10}\}$	$\{3V_{ES_1}, p_2\}$	$\{2V_{ES_1}, p_1, p_9\}$	$\{t_1, t_7, 2t_{ES_1}\}$	$\{t_2, t_6, 3t_{ES_1}\}$
ES_2	$\{p_3\}, \{p_8\}\surd$ PR$_1$, PR$_2\surd$	$\{p_1\}, \{p_5\}\surd$	$\{p_{10}\}, \{p_{11}\}\surd$	$\{3V_{ES_2}, p_8\}$	$\{2V_{ES_2}, p_5, p_{11}\}$	$\{t_2, t_8, 2t_{ES_2}\}$	$\{t_3, t_7, 3t_{ES_2}\}$

CP: control place; CT: control transition.
$\surd$ denotes the selected production routes PR$_m$($m = 1, 2$), the corresponding idle, operation, and resource places, respectively;
$M_0(V_{ES_1}) = M_0(V_{ES_2}) = 0$.

Table 3. Controllability test to one dependent siphon.

j	DS_j	$M'(DS_j)$	$M'([DS_j])$	Controllability
1	DS_1	2	2	No

Table 4. The supplemental addition of CPs and CTs to two elementary siphons and the addition of CP and CT to one dependent siphon.

ES_i	$p_{ES_i} \in {}^\bullet(V_{ES_i}{}^\bullet) \cap PR_m$	p_1^0, p_2^0	r_{ES_i}	$^\bullet t_{ES_i}$	$t_{ES_i}{}^\bullet$	$^\bullet V_{ES_i}$	$V_{ES_i}{}^\bullet$
DS_i	$p_{DS_i} \in {}^\bullet(V_{DS_i}{}^\bullet) \cap PR_m$	p_1^0, p_2^0	r_{DS_i}	$^\bullet t_{DS_i}$	$t_{DS_i}{}^\bullet$	$^\bullet V_{DS_i}$	$V_{DS_i}{}^\bullet$
ES_1	$\{p_2\}\surd, \{p_7\}$ PR$_1\surd$, PR$_2$	$\{p_1\}\surd, \{p_5\}$	$\{p_9\}\surd, \{p_{10}\}$	$\{3V_{ES_1}, p_2, V_{DS_1}\}$	$\{2V_{ES_1}, p_1, p_9\}$	$\{t_1, t_7, 2t_{ES_1}\}$	$\{t_2, t_6, 3t_{ES_1}\}$
ES_2	$\{p_3\}, \{p_8\}\surd$ PR$_1$, PR$_2\surd$	$\{p_1\}, \{p_5\}\surd$	$\{p_{10}\}, \{p_{11}\}\surd$	$\{3V_{ES_2}, p_8, V_{DS_1}\}$	$\{2V_{ES_2}, p_5, p_{11}\}$	$\{t_2, t_8, 2t_{ES_2}\}$	$\{t_3, t_7, 3t_{ES_2}, t_{DS_1}\}$
DS_1	$\{p_3\}\surd, \{p_7\}$ PR$_1\surd$, PR$_2$	$\{p_1\}\surd, \{p_5\}$	$\{p_{10}\}\surd, \{p_{10}\}$	$\{4V_{DS_1}, p_3, V_{ES_2}\}$	$\{3V_{DS_1}, p_1, p_{10}\}$	$\{t_1, t_8, 3t_{DS_1}\}$	$\{t_3, t_6, 4t_{DS_1}, t_{ES_1}, t_{ES_2}\}$

CP: control place; CT: control transition.
$\surd$ denotes the selected production routes PR$_m$($m = 1, 2$), the corresponding idle, operation, and resource places, respectively; $M'(V_{DS_1}) = 0$.

Table 5. A two-stage deadlock control process using Algorithm 1 for a marked S^{3}PR shown in Figure 1.

The state of Petri net system	No. of DS	No. of DFS
An original system, (N_0, M_0)	5	42※
An extended system, (N', M')	1	46
A live controlled system, (N^*, M^*)	0	47*

DS: deadlock states; DFS: deadlock-free states.

Table 6. Comparison of the proposed DCP with those DCPs in the literature.[22,25,29]

Criteria	Algorithm 1	*ECM*	*PCF*	*LZ*
No. of added CPs	3	3	3	2
No. of added CTs	3	0	0	0
No. of reachable states	47*	36	42※	36
R	100%	76.6%	89.4%	76.6%

CP: control place; CT: control transition; DCP: deadlock control policy.

to DSs, a two-stage deadlock control process using Algorithm 1, and its control performance comparison with the existing approaches, respectively.

Note that this dependent siphon meets the controllability condition from Table 9, so it is not needed to add a CP and CT to DS_1 due to Proposition 1. Algorithm 1 outputs a live controlled system (N^*, M^*), that is, (N^*, M^*) is the same as (N', M').

Table 7. Two elementary siphons and one dependent siphon, their complementary sets, and presets and postsets of the complementary sets.

i	ES_i	$[ES_i]$	$^\bullet[ES_i]$	$[ES_i]^\bullet$
j	DS_j	$[DS_j]$	$^\bullet[DS_j]$	$[DS_j]^\bullet$
1	$\{p_5, p_6, p_{11}, p_{14}, p_{15}\}$	$\{p_4, p_{10}\}$	$\{t_6, t_{11}\}$	$\{t_7, t_{10}\}$
2	$\{p_7, p_{10}, p_{15}, p_{16}\}$	$\{p_5, p_6, p_9\}$	$\{t_3, t_7, t_{12}\}$	$\{t_4, t_8, t_{11}\}$
1	$\{p_7, p_{11}, p_{14}, p_{15}, p_{16}\}$	$\{p_4, p_5, p_6, p_9, p_{10}\}$	$\{t_3, t_6, t_{12}\}$	$\{t_4, t_8, t_{10}\}$

Table 8. The addition of CPs and CTs to two elementary siphons.

ES_i	$p_{ES_i} \in {}^\bullet(V_{ES_i}{}^\bullet) \cap PR_m$	p_1^0, p_2^0	r_{ES_i}	$^\bullet t_{ES_i}$	$t_{ES_i}{}^\bullet$	$^\bullet V_{ES_i}$	$V_{ES_i}{}^\bullet$
ES_1	$\{p_4\}\surd$, $\{p_{10}\}$ PR$_1$ $\surd$PR$_2$	$\{p_1\}\surd$, $\{p_8\}$	$\{p_{14}\}\surd$, $\{p_{15}\}$	$\{3V_{ES_1}, p_4\}$	$\{2V_{ES_1}, p_1, p_{14}\}$	$\{t_6, t_{11}, 2t_{ES_1}\}$	$\{t_7, t_{10}, 3t_{ES_1}\}$
ES_2	$\{p_5, p_6\}$, $\{p_9\}\surd$ PR$_1$, PR$_2\surd$	$\{p_1\}$, $\{p_8\}\surd$	$\{p_{15}\}$, $\{p_{16}\}\surd$	$\{2V_{ES_2}, p_9\}$	$\{V_{ES_2}, p_8, p_{16}\}$	$\{t_3, t_7, t_{12}, t_{ES_2}\}$	$\{t_4, t_8, t_{11}, 2t_{ES_2}\}$

CP: control place; CT: control transition.
$\surd$ denotes the selected production routes PR$_m$($m = 1, 2$), the corresponding idle, operation, and resource places, respectively;
$M_0(V_{ES_1}) = M_0(V_{ES_2}) = 0$.

Table 9. Controllability test to one dependent siphon.

j	DS_j	$M'(DS_j)$	$M'([DS_j])$	Controllability
1	DS_1	3	1	Yes

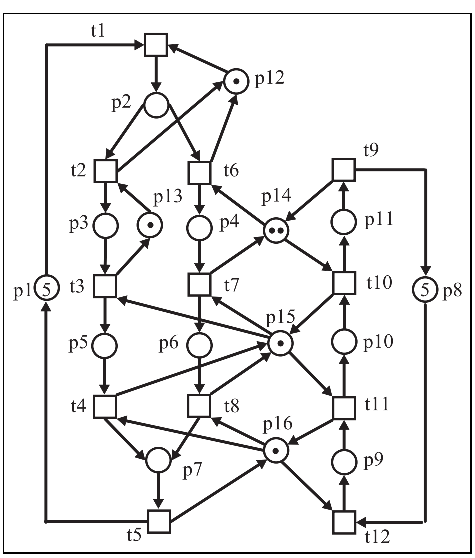

Figure 3. A non-live marked ES^{3}PR.

Finally, Figure 4 shows a well-known S^{3}R model with deadlocks,[18,22,23,25,29,31,38,45] where $P^0 = \{p_1, p_5, p_{14}\}$, $P_A = \{p_2 - p_4, \ -p_6 - p_{13}, \ p_{15}, \ -p_{19}\}$, $P_R = \{p_{20} - p_{26}\}$. We have $|P| = |P^0| + |P_A| + |P_R| = 26$, $|T| = 20$, and three production routes PR$_m$($m = 1, 2, 3$), respectively.

Table 10. A two-stage deadlock control process using Algorithm 1 for a marked ES^{3}PR shown in Figure 3.

The state of Petri net system	No. of DS	No. of DFS
An original system, (N_0, M_0)	56	194※
An extended system, (N', M')	0	250*
A live controlled system, (N^*, M^*)	0	250*

DS: deadlock states; DFS: deadlock-free states.

Tables 12–17 show solution of ESs, DSs, and their complementary sets, the addition of CPs and CTs to ESs, controllability test to DSs, the supplemental addition of CPs and CTs to ESs and the addition of CPs and CTs to DSs, a two-stage deadlock control process using Algorithm 1, and its control performance comparison with the existing approaches, respectively.

From Table 17, it is seen that $C2$ adds seven CPs and CTs to an S^{3}R example depicted in Figure 4 so that a live controlled Petri net system with MRN (26,750) is achieved as well, which is the same as Algorithm 1. However, it is unclear for $C2$ to select 7 siphons from

Table 11. Comparison of the proposed DCP with those DCPs in the literature.[25,29,31,41]

Criteria	Algorithm 1	*ECM*	*HJX*$^+$	*PCF*	*LAW*$^+$
No. of added CPs	2	3	4	3	3
No. of added CTs	2	0	0	0	0
No. of reachable states	250*	49	156	194※	194※
R	100%	19.6%	62.4%	77.6%	77.6%

CP: control place; CT: control transition; DCP: deadlock control policy.

Table 12. Six elementary siphons and 12 dependent siphons, their complementary sets, and presets and postsets of the complementary sets.

i	ES_i	$[ES_i]$	$^\bullet[ES_i]$	$[ES_i]^\bullet$
j	DS_j	$[DS_j]$	$^\bullet[DS_j]$	$[DS_j]^\bullet$
1	$\{p_4, p_9, p_{12}, p_{17}, p_{21}, p_{24}\}$	$\{p_2, p_3, p_8\}$	$\{t_3, t_{11}\}$	$\{t_4, t_{13}\}$
2	$\{p_4, p_{10}, p_{17}, p_{21}, p_{22}, p_{24}, p_{26}\}$	$\{p_2, p_3, p_8, p_9, p_{12}, p_{13}, p_{18}, p_{19}\}$	$\{t_3, t_8, t_{11}, t_{15}\}$	$\{t_5, t_{10}, t_{13}, t_{17}\}$
3	$\{p_2, p_4, p_8, p_{12}, p_{16}, p_{21}, p_{25}\}$	$\{p_{11}, p_{17}\}$	$\{t_7, t_{17}\}$	$\{t_8, t_{18}\}$
4	$\{p_2, p_4, p_8, p_{13}, p_{17}, p_{21}, p_{26}\}$	$\{p_{12}, p_{18}\}$	$\{t_8, t_{16}\}$	$\{t_9, t_{17}\}$
5	$\{p_2, p_4, p_8, p_{12}, p_{15}, p_{20}, p_{21}, p_{23}, p_{25}\}$	$\{p_6, p_7, p_{11}, p_{16}, p_{17}\}$	$\{t_1, t_{17}\}$	$\{t_3, t_8, t_{19}\}$
6	$\{p_{10}, p_{18}, p_{22}, p_{26}\}$	$\{p_{13}, p_{19}\}$	$\{t_9, t_{15}\}$	$\{t_{10}, t_{16}\}$
1	$\{p_4, p_{10}, p_{15}, p_{20} - p_{26}\}$	$\{p_2, p_3, p_6 - p_8, p_{11} - p_{13}, p_{16} - p_{19}\}$	$\{t_1, t_7, t_{11}, t_{15}\}$	$\{t_4, t_{10}, t_{13}, t_{19}\}$
2	$\{p_4, p_{10}, p_{16}, p_{21}, p_{22}, p_{24} - p_{26}\}$	$\{p_2, p_3, p_8, p_9, p_{11} - p_{13}, p_{17} - p_{19}\}$	$\{t_3, t_7, t_{11}, t_{15}\}$	$\{t_5, t_{10}, t_{13}, t_{18}\}$
3	$\{p_4, p_9, p_{13}, p_{15}, p_{20}, p_{21}, p_{23} - p_{26}\}$	$\{p_2, p_3, p_6 - p_8, p_{11}, p_{12}, p_{16} - p_{18}\}$	$\{t_1, t_7, t_{11}, t_{16}\}$	$\{t_4, t_9, t_{13}, t_{19}\}$
4	$\{p_4, p_9, p_{13}, p_{16}, p_{21}, p_{24} - p_{26}\}$	$\{p_2, p_3, p_8, p_{11}, p_{12}, p_{17}, p_{18}\}$	$\{t_3, t_7, t_{11}, t_{16}\}$	$\{t_4, t_9, t_{13}, t_{18}\}$
5	$\{p_4, p_9, p_{13}, p_{17}, p_{21}, p_{24}, p_{26}\}$	$\{p_2, p_3, p_6, p_{12}, p_{18}\}$	$\{t_1, t_8, t_{11}, t_{16}\}$	$\{t_2, t_7, t_9, t_{13}, t_{18}\}$
6	$\{p_4, p_9, p_{12}, p_{15}, p_{20}, p_{21}, p_{23} - p_{25}\}$	$\{p_2, p_3, p_6 - p_8, p_{11}, p_{16}, p_{17}\}$	$\{t_1, t_7, t_{11}, t_{18}\}$	$\{t_4, t_8, t_{13}, t_{19}\}$
7	$\{p_4, p_9, p_{12}, p_{16}, p_{21}, p_{24}, p_{25}\}$	$\{p_2, p_3, p_8, p_{11}, p_{17}\}$	$\{t_3, t_7, t_{11}, t_{17}\}$	$\{t_4, t_8, t_{13}, t_{18}\}$
8	$\{p_2, p_4, p_8, p_{10}, p_{15}, p_{20} - p_{23}, p_{25}, p_{26}\}$	$\{p_6, p_7, p_{11}, p_{13}, p_{16} - p_{19}\}$	$\{t_1, t_7, t_9, t_{15}\}$	$\{t_3, t_8, t_{10}, t_{19}\}$
9	$\{p_2, p_4, p_8, p_{13}, p_{15}, p_{20}, p_{21}, p_{23}, p_{25}, p_{26}\}$	$\{p_6, p_7, p_{11}, p_{12}, p_{16} - p_{18}\}$	$\{t_1, t_7, t_{16}\}$	$\{t_3, t_9, t_{19}\}$
10	$\{p_2, p_4, p_8, p_{10}, p_{16}, p_{21}, p_{22}, p_{25}, p_{26}\}$	$\{p_{11} - p_{13}, p_{17} - p_{19}\}$	$\{t_7, t_{15}\}$	$\{t_{10}, t_{18}\}$
11	$\{p_2, p_4, p_8, p_{13}, p_{16}, p_{21}, p_{25}, p_{26}\}$	$\{p_{11}, p_{12}, p_{17}, p_{18}\}$	$\{t_7, t_{16}\}$	$\{t_9, t_{18}\}$
12	$\{p_2, p_4, p_8, p_{10}, p_{17}, p_{21}, p_{22}, p_{26}\}$	$\{p_{12}, p_{13}, p_{18}, p_{19}\}$	$\{t_8, t_{15}\}$	$\{t_{10}, t_{17}\}$

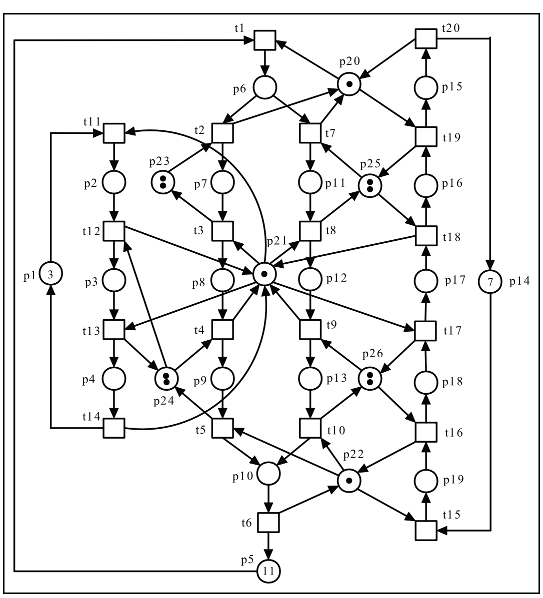

Figure 4. A large-sized marked S^{3}R with deadlocks.

18 siphons; in other words, the relevant rule on how to select those siphons that are needed the addition of CPs and CTs from all solved siphons is not given in the study of Chao.[40] On the contrary, after dividing 18 siphons into 6 ESs and 12 DSs and finishing the controllability test for all DSs, Algorithm 1 decisively adds 6 CPs and CTs for 6 ESs and 1 CP and CT for 1 DS that does not meet controllability, which is performed in two stages. As stated above, Algorithm 1 can make all ESs and a few of DSs explicitly controlled by the addition of CPs and CTs while the remaining DSs that meet controllability are implicitly controlled. Therefore, Algorithm 1 may reduce the SC of the finally live controlled system (N^*, M^*) with MRN to some degree compared with the corresponding (N^*, M^*) with MRN obtained by the study of Chao.[40]

Conclusion

It is a challenge to design the effective deadlock control policy (DCPs) by means of the relevant theories and applications of Petri nets. Motivated by the advantages of ESs[14,22,23] and CPs and CTs,[40] a two-stage DCA is proposed in this article. Unlike these DPPs with NMPB in the existing literature, it can obtain a live controlled system (N^*, M^*) with MRN. In addition, by

Table 13. The addition of CPs and CTs to six elementary siphons.

ES_i	$p_{ES_i} \in {}^\bullet(V_{ES_i}{}^\bullet) \cap PR_m$	p_1^0, p_2^0	r_{ES_i}	${}^\bullet t_{ES_i}$	$t_{ES_i}{}^\bullet$	${}^\bullet V_{ES_i}$	$V_{ES_i}{}^\bullet$
ES_1	$\{p_3\}\surd, \{p_8\}$ $PR_1\surd$, PR_2, PR_3	$\{p_1\}\surd, \{p_5\}, \{p_{14}\}$	$\{p_{24}\}\surd, \{p_{21}\}$	$\{3V_{ES_1}, p_3, V_{ES_2}\}$	$\{2V_{ES_1}, p_1, p_{24}\}$	$\{t_3, t_{11}, 2t_{ES_1}\}$	$\{t_4, t_{13}, 3t_{ES_1}\}$
ES_2	$\{p_3\}, \{p_9\}\surd, \{p_{13}\}, \{p_{18}\}$ PR_1, $PR_2\surd$, PR_3	$\{p_1\}, \{p_5\}\surd, \{p_{14}\}$	$\{p_{24}\}\surd, \{p_{26}\}$	$\{6V_{ES_2}, p_9\}$	$\{5V_{ES_2}, p_5, p_{24}\}$	$\{t_3, t_8, t_{11}, t_{15}, 5t_{ES_2}\}$	$\{t_5, t_{10}, t_{13}, t_{17}, 6t_{ES_2}, t_{ES_1}, t_{ES_4}, t_{ES_6}\}$
ES_3	$\{p_{11}\}, \{p_{17}\}\surd$ PR_1, PR_2, $PR_3\surd$	$\{p_1\}, \{p_5\}, \{p_{14}\}\surd$	$\{p_{25}\}, \{p_{21}\}\surd$	$\{3V_{ES_3}, p_{17}, V_{ES_5}\}$	$\{2V_{ES_3}, p_{14}, p_{21}\}$	$\{t_7, t_{17}, 2t_{ES_3}\}$	$\{t_8, t_{18}, 3t_{ES_3}\}$
ES_4	$\{p_{12}\}\surd, \{p_{18}\}$ PR_1, $PR_2\surd$, PR_3	$\{p_1\}, \{p_5\}\surd, \{p_{14}\}$	$\{p_{21}\}\surd, \{p_{26}\}$	$\{3V_{ES_4}, p_{12}, V_{ES_2}\}$	$\{2V_{ES_4}, p_5, p_{21}\}$	$\{t_8, t_{16}, 2t_{ES_4}\}$	$\{t_9, t_{17}, 3t_{ES_4}\}$
ES_5	$\{p_7\}, \{p_{11}\}, \{p_{16}\}\surd$ PR_1, PR_2, $PR_3\surd$	$\{p_1\}, \{p_5\}, \{p_{14}\}\surd$	$\{p_{23}\}, \{p_{25}\}\surd$	$\{6V_{ES_5}, p_{16}\}$	$\{5V_{ES_5}, p_{14}, p_{25}\}$	$\{t_1, t_{17}, 5t_{ES_5}\}$	$\{t_3, t_8, t_{19}, 6t_{ES_5}, t_{ES_3}\}$
ES_6	$\{p_{13}\}\surd, \{p_{19}\}$ PR_1, $PR_2\surd$, PR_3	$\{p_1\}, \{p_5\}\surd, \{p_{14}\}$	$\{p_{26}\}\surd, \{p_{22}\}$	$\{3V_{ES_6}, p_{13}, V_{ES_2}\}$	$\{2V_{ES_6}, p_5, p_{26}\}$	$\{t_9, t_{15}, 2t_{ES_6}\}$	$\{t_{10}, t_{16}, 3t_{ES_6}\}$

CP: control place; CT: control transition.
$\surd$ denotes the selected production routes $PR_m (m = 1, 2, 3)$, the corresponding idle, operation, and resource places, respectively;
$M_0(V_{ES_1}) = M_0(V_{ES_2}) = \cdots = M_0(V_{ES_6}) = 0$.

Table 14. Controllability test to 12 dependent siphons.

j	DS_j	$M'(DS_j)$	$M'([DS_j])$	Controllability
1	DS_1	8	3	Yes
2	DS_2	6	2	Yes
3	DS_3	6	4	Yes
4	DS_4	6	1	Yes
5	DS_5	4	1	Yes
6	DS_6	5	3	Yes
7	DS_7	5	0	Yes
8	DS_8	8	1	Yes
9	DS_9	5	3	Yes
10	DS_{10}	4	2	Yes
11	DS_{11}	4	1	Yes
12	DS_{12}	1	3	No

constructing an IPP of P-invariants of (N', M') to perform the controllability test for all DSs in N', it can also simplify the structure of (N^*, M^*) to some degree comparing with the deadlock control policy.[40] A theoretical analysis and several examples that belong to S^3PR and ES^3PR in the existing literature are used to illustrate efficiency of the proposed DCA. In essence, this two-stage DCA belongs to the combination of DPP and deadlock detection and recovery policy (DDRP) and possesses some features with both of them. The focus of our future research is to extend this DCA to more general classes of Petri nets such as S^4R and G-system[1] by modifying the weight values of control arcs of CTs linking to corresponding idle places, resource places, operation places, and CPs.

Table 15. The supplemental addition of CPs and CTs to six elementary siphons and the addition of CP and CT to one dependent siphon.

ES_i / DS_i	$p_{ES_i} \in {}^\bullet(V_{ES_i}{}^\bullet) \cap PR_m$ / $p_{DS_i} \in {}^\bullet(V_{DS_i}{}^\bullet) \cap PR_m$	p_1^0, p_2^0 / p_1^0, p_2^0	r_{ES_i} / r_{DS_i}	${}^\bullet t_{ES_i}$ / ${}^\bullet t_{DS_i}$	$t_{ES_i}{}^\bullet$ / $t_{DS_i}{}^\bullet$	${}^\bullet V_{ES_i}$ / ${}^\bullet V_{DS_i}$	$V_{ES_i}{}^\bullet$ / $V_{DS_i}{}^\bullet$
ES_1	$\{p_3\}\surd, \{p_8\}$ $PR_1\surd$, PR_2, PR_3	$\{p_1\}\surd$, $\{p_5\}$, $\{p_{14}\}$	$\{p_{24}\}\surd$, $\{p_{21}\}$	$\{3V_{ES_1}, p_3, V_{ES_2}\}$	$\{2V_{ES_1}, p_1, p_{24}\}$	$\{t_3, t_{11}, 2t_{ES_1}\}$	$\{t_4, t_{13}, 3t_{ES_1}\}$
ES_2	$\{p_3\}, \{p_9\}\surd, \{p_{13}\}, \{p_{18}\}$ PR_1, $PR_2\surd$, PR_3	$\{p_1\}$, $\{p_5\}\surd$, $\{p_{14}\}$	$\{p_{24}\}\surd$, $\{p_{26}\}$	$\{6V_{ES_2}, p_9\}$	$\{5V_{ES_2}, p_5, p_{24}\}$	$\{t_3, t_8, t_{11}, t_{15}, 5t_{ES_2}\}$	$\{t_5, t_{10}, t_{13}, t_{17}, 6t_{ES_2}, t_{ES_1}, t_{ES_4}, t_{ES_6}, t_{DS_1}\}$
ES_3	$\{p_{11}\}, \{p_{17}\}\surd$ PR_1, PR_2, $PR_3\surd$	$\{p_1\}$, $\{p_5\}$, $\{p_{14}\}\surd$	$\{p_{25}\}$, $\{p_{21}\}\surd$	$\{3V_{ES_3}, p_{17}, V_{ES_5}\}$	$\{2V_{ES_3}, p_{14}, p_{21}\}$	$\{t_7, t_{17}, 2t_{ES_3}\}$	$\{t_8, t_{18}, 3t_{ES_3}\}$
ES_4	$\{p_{12}\}\surd, \{p_{18}\}$ PR_1, $PR_2\surd$, PR_3	$\{p_1\}$, $\{p_5\}\surd$, $\{p_{14}\}$	$\{p_{21}\}\surd$, $\{p_{26}\}$	$\{3V_{ES_4}, p_{12}, V_{ES_2}, V_{DS_1}\}$	$\{2V_{ES_4}, p_5, p_{21}\}$	$\{t_8, t_{16}, 2t_{ES_4}\}$	$\{t_9, t_{17}, 3t_{ES_4}\}$
ES_5	$\{p_7\}, \{p_{11}\}, \{p_{16}\}\surd$ PR_1, PR_2, $PR_3\surd$	$\{p_1\}$, $\{p_5\}$, $\{p_{14}\}\surd$	$\{p_{23}\}$, $\{p_{25}\}\surd$	$\{6V_{ES_5}, p_{16}\}$	$\{5V_{ES_5}, p_{14}, p_{25}\}$	$\{t_1, t_{17}, 5t_{ES_5}\}$	$\{t_3, t_8, t_{19}, 6t_{ES_5}, t_{ES_3}\}$
ES_6	$\{p_{13}\}\surd, \{p_{19}\}$ PR_1, $PR_2\surd$, PR_3	$\{p_1\}$, $\{p_5\}\surd$, $\{p_{14}\}$	$\{p_{26}\}\surd$, $\{p_{22}\}$	$\{3V_{ES_6}, p_{13}, V_{ES_2}, V_{DS_1}\}$	$\{2V_{ES_6}, p_5, p_{26}\}$	$\{t_9, t_{15}, 2t_{ES_6}\}$	$\{t_{10}, t_{16}, 3t_{ES_6}, t_{DS_1}\}$
DS_1	$\{p_{13}\}\surd, \{p_{18}\}$ PR_1, $PR_2\surd$, PR_3	$\{p_1\}$ $\{p_5\}\surd$, $\{p_{14}\}$	$\{p_{26}\}\surd$, $\{p_{26}\}$	$\{4V_{DS_1}, p_{13}, V_{ES_2}, V_{ES_6}\}$	$\{3V_{DS_1}, p_5, p_{26}\}$	$\{t_8, t_{15}, 3t_{DS_1}\}$	$\{t_{10}, t_{17}, 4t_{DS_1}, t_{ES_4}, t_{ES_6}\}$

CP: control place; CT: control transition.
$\surd$ denotes the selected production routes $PR_m (m = 1, 2, 3)$, the corresponding idle, operation, and resource places, respectively; $M'(V_{DS_1}) = 0$.

Table 16. A two-stage deadlock control process using Algorithm 1 for a marked S^3PR shown in Figure 4.

The state of Petri net system	No. of DS	No. of DFS
An original system, (N_0, M_0)	4226	22524※
An extended system, (N', M')	234	26,516
A live controlled system, (N^*, M^*)	0	26,750*

DS: deadlock states; DFS: deadlock-free states.

Table 17. Comparison of the proposed DCP with those DCPs in the previous studies.[18,22,25,29,31,38,40,45]

Criteria	Algorithm 1	*ECM*	*LZ*	*HJX*$^+$	*UZ*	*PCF*	*HHJ*$^+$	*C1*	*C2*
No. of added CPs	7	18	5	7	19	13	9	7	7
No. of added CTs	7	0	0	0	0	0	0	0	7
No. of reachable states	26, 750*	6287	15,999	16,425	21,562	21,581	20,444	21,585	26, 750*
R	100%	23.5%	59.8%	61.4%	80.6%	80.7%	70.4%	80.7%	100%
Computational complexity	NP-hard	Exponential	Exponential	Exponential	Exponential	Exponential	NP-hard	Exponential	NP-hard

CP: control place; CT: control transition; NP: nondeterministic polynomial time; DCP: deadlock control policy.

Acknowledgements

The authors thank the editor and anonymous reviewers whose comments and suggestions greatly helped to improve the quality and presentation of this paper.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Natural Science Foundation of China (NSFC, grant no. 61364004), Chinese Visiting Scholars to Study Overseas Program sponsored by China Scholarship Council Foundation (grant no. 201408625045), the Scientific Research Program funded by Shaanxi Provincial Education Department (grant no. 16JK1342), the Doctoral Research funds of Lanzhou University of Technology (grant no. 04-237), and Alumni Foundation Civil Engineering 77, Lanzhou University of Technology (grant no. TM-QK1301).

References

1. Li ZW and Zhou MC. *Deadlock resolution in automated manufacturing systems*. 1st ed. Berlin: Springer, 2009.
2. Chen YF, Li ZW, Barkaoui K, et al. Compact supervisory control of discrete event systems by Petri nets with data inhibitor arcs. *IEEE T Syst Man Cy: S* 2017; 47: 364–379.
3. Huang YS, Jeng MD, Xie XL, et al. Deadlock prevention policy based on Petri nets and siphons. *Int J Prod Res* 2001; 39: 283–305.
4. Park J and Reveliotis SA. Deadlock avoidance in sequential resource allocation systems with multiple resource acquisitions and flexible routings. *IEEE T Automat Contr* 2001; 46: 1572–1583.
5. Tong Y, Li ZW and Guia A. On the equivalence of observation structures for Petri net generators. *IEEE T Automat Contr* 2016; 61: 2448–2462.
6. Uzam M and Li ZW. On an iterative deadlock prevention approach for automated manufacturing systems. *Int J Adv Manuf Tech* 2014; 74: 503–507.
7. Chao DY. Direct minimal empty siphon computation using MIP. *Int J Adv Manuf Tech* 2009; 45: 397–405.
8. Li SY, Li ZW and Hu HS. Siphon extraction for deadlock control in flexible manufacturing systems by using Petri nets. *Int J Comp Integ M* 2011; 24: 710–725.
9. Li ZW and Zhou MC. Control of elementary and dependent siphons in Petri nets and their application. *IEEE T Syst Man Cy A* 2008; 38: 133–148.
10. Liu HX, Wu WM, Su HY, et al. Design of optimal Petri-net controllers for a class of flexible manufacturing systems with key resources. *Inform Sciences* 2016; 363: 221–234.
11. Huang B, Zhou MC and Zhang GX. Synthesis of Petri net supervisors for FMS via redundant constraint elimination. *Automatica* 2015; 61: 156–163.
12. Liu GY and Barkaoui K. A survey of siphons in Petri nets. *Inform Sciences* 2016; 363: 198–220.
13. Chen YF, Li ZW, Barkaoui K, et al. New Petri net structure and its application to optimal supervisory control: interval inhibitor arcs. *IEEE T Syst Man Cy: S* 2014; 44: 1384–1400.
14. Chao DY and Chen JT. Segment theory to compute elementary siphons in Petri nets for deadlock control. *J Chin Inst Ind Eng* 2011; 28: 573–585.
15. Chen YF and Li ZW. Design of a maximally permissive liveness-enforcing supervisor with a compressed supervisory structure for flexible manufacturing systems. *Automatica* 2011; 47: 1028–1034.
16. Chen YF, Li ZW, Barkaoui K, et al. On the enforcement of a class of nonlinear constraints on Petri nets. *Automatica* 2015; 55: 116–124.

17. Chen YF, Li ZW, Wu NQ, et al. Deadlock recovery for flexible manufacturing systems modeled with Petri nets. *Inform Sciences* 2017; 381: 290–303.
18. Hong L, Hou YF, Jing JF, et al. Deadlock prevention policy with behavioral optimality or suboptimality achieved by the redundancy identification of constraints and the rearrangement of monitors. *Discrete Dyn Nat Soc* 2015; 2015: 579623.
19. Hou YF, Li ZW, Zhao M, et al. Extended elementary siphon-based deadlock prevention policy for a class of generalised Petri nets. *Int J Comp Integ M* 2014; 27: 85–102.
20. Huang YS. Design of deadlock prevention supervisors using Petri nets. *Int J Adv Manuf Tech* 2007; 35: 349–362.
21. Li SY and Li ZW. Structure reduction of liveness-enforcing Petri nets using mixed integer programming. *Asian J Control* 2012; 14: 384–399.
22. Li ZW and Zhou MC. Elementary siphons of Petri nets and their application to deadlock prevention in flexible manufacturing systems. *IEEE T Syst Man Cy A* 2004; 34: 38–51.
23. Li ZW and Zhou MC. Two-stage method for synthesizing liveness-enforcing supervisors for flexible manufacturing systems using Petri nets. *IEEE T Ind Inform* 2006; 2: 313–325.
24. Li ZW, Hu HS and Wang AR. Design of liveness-enforcing supervisors for flexible manufacturing systems using Petri nets. *IEEE T Syst Man Cy C* 2007; 37: 517–526.
25. Piroddi L, Cordon R and Fumagalli I. Combined siphon and marking generation for deadlock prevention in Petri nets. *IEEE T Syst Man Cy A* 2009; 39: 650–661.
26. Tong Y, Li ZW, Seatzu C, et al. Verification of state-based opacity using Petri nets. *IEEE T Automat Contr* 2017; 62: 2823–2837.
27. Li ZW and Zhao M. On controllability of dependent siphons for deadlock prevention in generalized Petri nets. *IEEE T Syst Man Cy A* 2008; 38: 369–384.
28. Li ZW, Wu NQ and Zhou MC. Deadlock control of automated manufacturing systems based on Petri nets—a literature review. *IEEE T Syst Man Cy C* 2012; 42: 437–462.
29. Ezpeleta J, Colom JM and Martinez J. A Petri net based deadlock prevention policy for flexible manufacturing systems. *IEEE T Robotic Autom* 1995; 11: 173–184.
30. Hou YF, Zhao M, Liu D, et al. An efficient siphon-based deadlock prevention policy for a class of generalized Petri nets. *Discrete Dyn Nat Soc* 2016; 2016: 8219424.
31. Huang YS, Jeng MD, Xie XL, et al. Siphon-based deadlock prevention policy for flexible manufacturing systems. *IEEE T Syst Man Cy A* 2006; 36: 1248–1256.
32. Huang B, Zhou MC, Zhang GX, et al. On further reduction of constraints in "nonpure Petri net supervisors for optimal deadlock control of flexible manufacturing systems." *IEEE T Syst Man Cy A* 2015; 45: 542–543.
33. Li SY, An AM, Wang Y, et al. Design of liveness-enforcing supervisors with simpler structures for deadlock-free operations in flexible manufacturing systems using necessary siphons. *J Intell Manuf* 2013; 24: 1157–1173.
34. Li SY, An AM, Cai Y, et al. A deadlock control policy for a subclass of Petri nets G–system. *Control Theory Appl* 2013; 30: 1329–1336.
35. Li SY, Li ZW, Hu HS, et al. An extraction algorithm for a set of elementary siphons based on mixed-integer programming. *J Syst Sci Syst Eng* 2012; 21: 106–125.
36. Chen YF, Li ZW and Zhou MC. Behaviorally optimal and structurally simple liveness-enforcing supervisors of flexible manufacturing systems. *IEEE T Syst Man Cy A* 2012; 42: 615–629.
37. Chen YF, Li ZW and Zhou MC. Optimal supervisory control of flexible manufacturing systems by Petri nets: a set classification approach. *IEEE T Autom Sci Eng* 2014; 11: 549–563.
38. Uzam M and Zhou MC. An improved iterative synthesis method for liveness enforcing supervisors of flexible manufacturing systems. *Int J Prod Res* 2006; 44: 1987–2030.
39. Huang B, Zhou MC, Zhang GX, et al. Lexicographic multiobjective integer programming for optimal and structurally minimal Petri net supervisors of automated manufacturing systems. *IEEE T Syst Man Cy A* 2015; 45: 1459–1470.
40. Chao DY. A new optimal control policy for a well-known S^3PR (systems of simple sequential processes with resources). *Int J Prod Res* 2012; 50: 1–13.
41. Li SY, An AM, Wu HM, et al. Policy to cope with deadlocks and livelocks for flexible manufacturing systems using the max'-controlled new smart siphons. *IET Control Theory Appl* 2014; 8: 1607–1616.
42. Integrated Net Analyzer (INA). A software tool for analysis of Petri nets (version 2.2), 2003, http://www.informatik.hu-berlin.de/starke/ina.html
43. LinGo. Premier optimization modeling tools, http://www.lingo.com/ (accessed October 2009).
44. Huang YS, Chun TH and Su PJ. Synthesis of deadlock prevention policy using Petri nets reachability graph technique. *Asian J Control* 2006; 12: 336–346.
45. Chao DY. Reaching more states for control of FMS. *Int J Prod Res* 2009; 48: 1217–1339.