Petri Net-based S^3PR Models of Automated Manufacturing Systems with Resources and Their Deadlock Prevention

František Čapkovič

Institute of Informatics, Slovak Academy of Sciences, Dúbravská cesta 9, 845 07 Bratislava, Slovakia, e-mail: Frantisek.Capkovic@savba.sk

Abstract: Correct allocation of resources in Automated Manufacturing Systems (AMS) is very important, especially in order to avoid deadlocks and their consequences. Petri Nets (PN) are frequently used for modeling AMS. S^3PR (Systems of Simple Sequential Processes with Resources) model of Resource Allocation Systems (RAS) based on PN are defined, analyzed and controlled here. S^3PR are modeled by Ordinary PN (OPN). After defining and creation of such models the deadlock prevention will be performed by two deadlock prevention methods, namely (i) the method based on elementary siphons, and (ii) the method based on preventing strict minimal siphons from being emptied in another way (by means of circuits, holders of resources and complementary siphons). For illustration, two practical examples will be introduced. Both approaches are very useful not only for reliable deadlock-free control of existing AMS, but also at design of new AMS of such kind.

Keywords: automated manufacturing systems; deadlock prevention; Petri nets; resource allocation systems; siphons; supervisor; traps

1 Introduction

Petri Nets (PN) in general are defined as follows. A Petri net is a four-tuple $N = (P, T, F, W)$, where P and T are finite nonempty sets. $P = \{p_1, p_2, \ldots, p_n\}$ is a set of places ($|P| = n$) and $T = \{t_1, t_2, \ldots, t_m\}$ is a set of transitions ($|T| = m$). It is valid that $P \cup T \neq \emptyset$ and $P \cap T = \emptyset$. $F = (P \times T) \cup (T \times \mathrm{P})$ is called a flow relation of the net. It is represented by directed arcs from places to transitions and from transitions to places. $W : (P \times T) \cup (T \times P) \longrightarrow \mathrm{N}$ is a mapping that assigns a weight to an arc: $W(f) > 0$ if $f \in F$ and $W(f) = 0$ otherwise. Here, $\mathrm{N} = \{0, 1, 2, \ldots\}$, containing natural numbers plus zero.

$N = (P, T, F, W)$ is *ordinary net* (OPN) denoted as $N = (P, T, F)$, if $\forall f \in F$, $W(f) = 1$. N is said to be a *generalized net* (GPN) if $\exists f \in F$, $W(f) > 1$. Consequently, PN marking M being an $(n \times 1)$ vector (frequently named also as the state vector) is evolved in the matrix/vector form as $M_{k+1} = M_k + [N].\sigma_k$, $k \in \mathrm{N}$, with M_0 being initial marking, where $[N] = [Post]^T - [Pre]$ is the $(n \times m)$ incidence matrix based on the set F, and σ_k is a $(m \times 1)$ firing vector of transitions (frequently named also as control vector). More details about PN can be found in [1]-[4]. While the foundations of PN were laid by C. A. Petri in his PhD Thesis [1], many other authors have developed the PN theory into its present form. Among them we should mention at least [2]-[4]. Some particulars about PN were mentioned also in [5]-[7].

A PN marking M is usually understand as a vector. $M(p)$ denotes the number of tokens in place p. For economy of space $\sum_{p \in P} M(p)p$ is used to denote the vector M. Place p is marked at M if $M(p) > 0$. A subset $S \subseteq P$ is marked (unmarked) at M if $M(S) > 0$ ($M(S) = 0$). If M_0 is an initial marking of a net N, (N, M_0) is called a *marked net*. A *state machine* is PN, where each transition has only one input and only one output place. PN where a place p is both an input and output place of a transition t is called *self-loop* PN. Here, in this paper, only PN without self-loops will be used.

2 Preliminaries

Resource allocation systems (RAS) represent a special class of automated manufacturing systems (AMS), where the attention is focused on resources. Resources are understood as a finite set of devices like robots, machine tools, automatically guided vehicles, transport belts, input/output devices, etc. Finite set of different processes of AMS (e.g. production lines) are competing each other for access to such resources. The competition may induce the existence of deadlocks. There exist several standard paradigms of RAS - see e.g. [8]-[11], where particulars about some of them are introduced, and [7], where a summary of most frequently used paradigms as well as their relation to PN in general are mentioned. In this paper, only one of them, namely S^3PR, modeled by ordinary Petri nets, will be presented and investigated. More complicated paradigms of RAS, e.g. extended S^3PR (ES^3PR) or S^4PR (Systems of Sequential Systems with Shared Process Resources), are modeled by means of generalized Petri nets. They will be investigated , the future.

2.1 S^3PR Model of AMS

A simple sequential process (S^2P) [7], where a review of definitions published in [12]-[14] are introduced, is a Petri net $N = (P_A \cup \{p^0\}, T, F)$, where

- $P_A \neq \emptyset$ is called the set of *activity places;* $\emptyset$ is an empty set
- $p^0 \notin P_A$ is called the idle *process place*
- N is a strongly connected *state machine*
- every circuit $\boldsymbol{C}$ of N contains place p^0.

A simple sequential process with resources (S^2PR) is a Petri net $N = (\{p^0\} \cup P_A \cup P_R, T, F)$ with P_R being a set of *resource places*, such that

- the subnet generated by $X = P_A \cup \{p^0\} \cup T$ is a S^2P
- $P_R \neq \emptyset$ and $(P_A \cup \{p^0\}) \cap P_R = \emptyset$
- $\forall t \in {}^\bullet p$, $\forall t' \in p^\bullet$, $\exists r_p \in P_R$, ${}^\bullet t \cap P_R = t'^\bullet \cap P_R = \{r_p\}$, where ${}^\bullet p$ expresses all input transitions of the place p, $p^\bullet$ represents all output transitions of p; ${}^\bullet t$ expresses all input places of the transition t, $t'^\bullet$ represents all output places of the transition t'
- the following statements are verified
    1. $\forall r \in P_R$, ${}^{\bullet\bullet}r \cap P_A = r^{\bullet\bullet} \cap P_A \neq \emptyset$
    2. $\forall r \in P_R$, ${}^\bullet r \cap r^\bullet = \emptyset$
- ${}^{\bullet\bullet}(p^0) \cap P_R = (p^0)^{\bullet\bullet} \cap P_R = \emptyset$

Here, ${}^\bullet r$ represent all input transitions of the resource place r, ${}^{\bullet\bullet}r = \bigcup_{t \in {}^\bullet r} {}^\bullet t$ is the set of all input places of all input transitions of the place r, $r^{\bullet\bullet} = \bigcup_{t \in r^\bullet} t^\bullet$ represents the set of all output places of all output transitions of the resource place r; ${}^{\bullet\bullet}(p^0)$ expresses all input places of all input transitions of the place p^0, $(p^0)^{\bullet\bullet}$ represents the set of all output places of all output transitions of the place p^0.

S^3PR N is composed of n S^2PR $N_1, N_2, ..., N_n$, i.e. $\text{O}_{i=1}^{n} N_i$.

2.1.1 Composition of Two S^2PR Into S^3PR

To illustrate the composition of two S^2PR N_1, N_2 into S^3PR N let us introduce the following.

An initial marking M_0 of S^2PR N is called an acceptable initial marking for N if

- $M_0(p^0) \geq 1$
- $M_0(p) = 0, \forall P_A$
- $M_0(r) \geq 1, \forall P_R$

S^2PR N with such a marking is said to be an *acceptable marked*.

In Simple Sequential Processes with Resources S^2PR N

- $P_A = P_{A1} \cup P_{A2}$
- $P^0 = \{p_1^0\} \cup \{p_2^0\}$
- $P_R = P_{R1} \cup P_{R2}$
- $T = T_1 \cup T_2$
- $F = F_1 \cup F_2$ is also S^3PR

S^3PR $N = N_1 \circ N_2$ (where $\circ$ symbolizes the composition of nets) is acceptably marked when

- $\forall i \in \{1,2\}, \forall p \in P_A \cup \{p_i^0\}, M_0(p) = M_{0i}(p)$
- $\forall i \in \{1,2\}, \forall r \in P_{Ri} \setminus P_C, M_0(r) = M_{0i}(r)$, where $P_C = P_{R1} \cap P_{R2} \neq \emptyset$
- $\forall r \in P_C, M_0(r) = \max\{M_{01}(r), M_{02}(r)\}$
- Places from P_A symbolize activities - a token in a place $p \in P_A$ models an active process - e.g. a part being processed. Places from P_R represent resources - e.g. a buffering capacity of resources, shared devices like robots and machines, etc. Tokens in a place $r \in P_R$ model the available buffering capacity of resource r. Markings represent states with a physical meaning. In this sense, only acceptable initial markings are considered. If the system is well defined and its initial marking is correct, all the markings that are reachable from it will represent possible states of the system and have physical meanings.

In Figure 1 we can see the S^2P N and two S^2PR N_1, N_2 nets, while their composition S^3PR net consisting of two S^2PR nets is displayed in Figure 2.

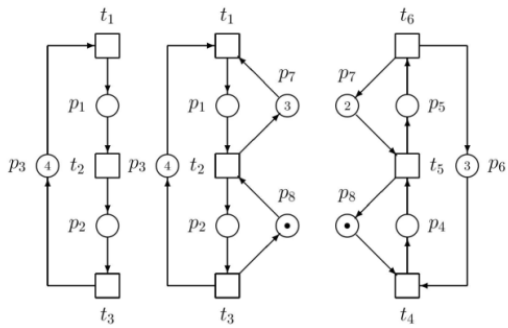

Figure 1
S^2P net (left) and two S^2PR nets (middle and right)

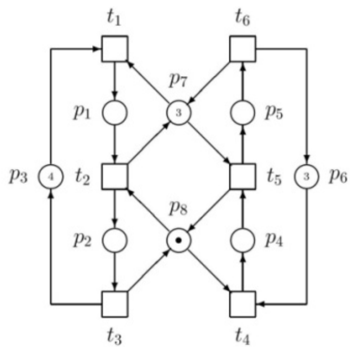

Figure 2
S^3PR net composed from two S^2PR nets N_1, N_2, i.e. $N = N_1 \circ N_2$

S^3PR in Figure 2 has the set of places $P^0 = \{p_1^0\} \cup \{p_2^0\} = \{p_3, p_6\}$, $P_A = P_{A1} \cup P_{A2} = \{p_1, p_2, p_4, p_5\}$, $P_R = P_{R1} \cup P_{R2} = \{p_7, p_8\}$. From Figure 2 is clear that ${}^\bullet p_7 = \{t_2, t_6\}$ and ${}^{\bullet\bullet}p_7 = {}^\bullet t_2 \cup {}^\bullet t_6 = \{p_1, p_5, p_8\}$, $p_7^\bullet = \{t_1, t_5\}$ and $p_7^{\bullet\bullet} = \{t_1^\bullet \cup t_5^\bullet\} = \{p_1, p_5, p_8\}$. Clearly, ${}^{\bullet\bullet}p_7 = p_7^{\bullet\bullet}$. In S^3PR, only one shared resource is allowed to be used at each stage in a job.

2.2 Deadlocks, Petri Net Siphons, Traps and *P*-Invariants

A *deadlock* in general is a state in which two or more processes are each waiting for the other one to execute, but neither can continue, , ,. Hence, deadlock is undesirable and rather bad phenomenon in PN models of real production processes.

There are four conditions for a deadlock occurring known as Coffman conditions [15]. A deadlock will never occur if one of these conditions is not satisfied. These conditions are the following:

1. There is a resource that cannot be used by more than one process at the same time (i.e. the mutual exclusion condition)
2. There are processes already holding resources are waiting for additional resources or may request new resources held by other processes (i.e. the hold and wait condition)
3. No resource can be forcibly removed from a process holding it. Resources can be released only by the explicit action of the process (i.e. not using a preemption condition)
4. Two or more processes form a circular chain where each process waits for a resource that the next process in the chain holds (the circular wait condition)

Petri net siphons, traps and invariants are structural PN parameters. They are intensively used at the deadlocks prevention. A nonempty subset $S \subset P$ is called a *siphon* if every transition having an output place in S has an input place in S. A nonempty subset $Q \subset P$ is called a *trap* if every transition having an input place in Q has an output place in Q.

If S has no token in a marking of N it remains without any token in each successor marking of N. If Q has at least one token in a marking of N it remains marked under each successor marking on N. If every non-empty siphon includes a marked trap, no dead marking is reachable. S is called an *empty siphon* at M_0 if $M_0(S) = \sum_{p \in S} M_0(p) = 0$. Such siphon is inclined to evocate deadlocks. The main aim of the deadlock prevention is the effort to prevent emptying siphons.

An $(n \times 1)$ vector I is the *P-invariant* (place invariant) if and only if $I \neq 0$ and $I^T \cdot [N] = 0^T$, where $[N]$ is the incidence matrix of N, 0 is zero vector. $\|I\| = \{p | I(\mathrm{p}) \neq 0\}$ is called the support of I. $\|I\|^+ = \{p | I(\mathrm{p}) > 0\}$ is the positive support of I.

2.2.1 Illustrative Example 1

The simple illustration of the siphon and trap is introduced in Figure 3. There, the siphon $S = \{p_2, p_4, p_5, p_6\}$ and the trap $Q = \{p_1, p_3, p_5, p_6\}$. Alike, in Figure 2 the siphon $S = \{p_2, p_5, p_7, p_8\}$ and the trap $Q = \{p_1, p_4, p_7, p_8\}$.

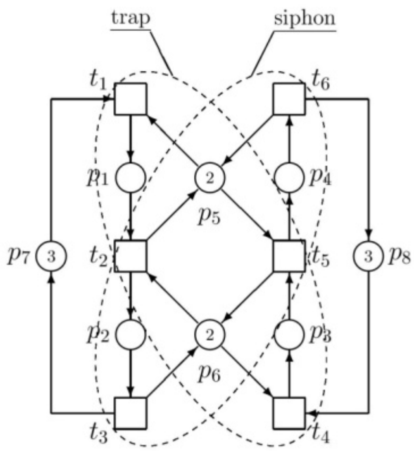

Figure 3
The siphon and trap in the S^{3}PR N

2.2.2 Minimal, Strict Minimal and Elementary Siphons

If a siphon does not properly contain another siphon, it is called a *minimal siphon*. The set of minimal siphons is denoted by Π. A minimal siphon S is called a *strict minimal siphon* (SMS) if there is no siphon contained in it as a proper subset. A strict minimal siphon is a siphon containing neither another siphon nor trap except itself.

Having a matrix $[\lambda]$ consisting of rows being strict minimal siphons, then the linearly independent rows of the matrix $[\eta] = [\lambda].[N]$ point out on *elementary siphons* in $[\lambda]$. Denote Π_E as a set of elementary siphons. In general - see [13], $|\Pi_E| \leq \min\{|P|, |T|\}$.

Other rows of $[\eta]$ point out on *dependent siphons* in $[\lambda]$. The dependent siphon may be *strict (strongly) dependent* or *slack (weakly) dependent.* It depends on whether the linear combination coefficients are all positive or not.

A siphon $S \notin \Pi_E$ is called *strongly dependent* siphon with respect to (w.r.t.) elementary siphons if $\eta_S = \sum_{Si \in \Pi_E} a_i.\eta_{Si}$, where $a_i \geq 0$. A siphon $S \notin \Pi_E$ is called *weakly dependent* siphon w.r.t. elementary siphons if $\exists A, B, A \neq \emptyset, B \neq \emptyset, A \cap B = \emptyset$ and $\eta_S = \sum_{Si \in A} a_i.\eta_{Si} - \sum_{Sj \in B} b_j.\eta_{Sj}$, where $a_i, b_j > 0$.

3 Deadlock Prevention in S3PR Models of RAS

There are several approaches to deadlock prevention [12]-[14], [16]-[23]. While in [12] elementary siphons are defined and their usage in the deadlock prevention is described, [13] is devoted to methods of the deadlock resolution in AMS. In [14] details of the supervisor synthesis for AMS are introduced. A survey of siphons is performed in [16], while a method of deadlock prevention without the need to enumerate complete set of siphons is presented in [17]. Different kinds of siphons, namely compound and complementary ones are analyzed in [18], while the control of elementary and dependent siphons is presented in [19]. The deadlock avoidance policy for AMS with assembly operations is described in [20]. Deadlock prevention methods depending on size of AMS are compared in [21]. Controllability for dependent siphons in S^3PR based on elementary siphons are tested in [22]. A practical usage of modeling and supervisory control in railway systems is presented in [23].

Two principled methods of deadlock prevention are presented and applied here on the S^3PR PN model of a real system and illustrated by examples: (i) the approach based on elementary siphons; and (ii) the approach based on preventing SMS from being emptied by means of analyzing circuits of PN models and using complementary siphons and downstream and upstream siphons.

3.1 Deadlock Prevention Method Based on Elementary Siphons

This approach is based on elementary siphons and siphons dependent on them. Both kinds of siphons were defined above in the part 2.2.2.

3.1.1 Illustrative Example

Consider an AMS schematically displayed in Figure 4.

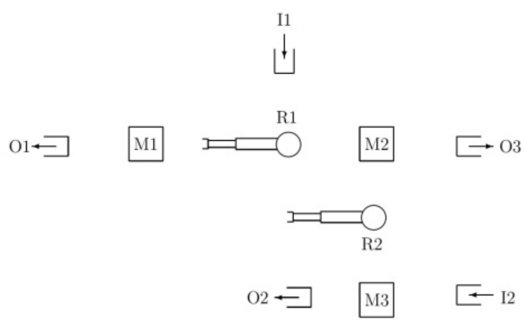

Figure 4
The scheme of the AMS structure

This AMS contains three machines M1 - M3, two robots R1, R2, two input devices I1, I2 and three output devices O1 - O3.

The scheme of the technological process being under way in it is displayed in Figure 5. Two types of parts P1 and P2 are processed as it is denoted by routing.

P1 is taken from I1 by R1 and put either into M1 or into M2. After processing P1

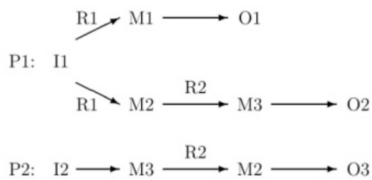

Figure 5
The scheme of the technological process of the AMS

by M1, P1 is moved to O1 by M1. After being loaded to M2, P1 is processed by M2 and then moved from M2 to M3 by R2. After being processed by M3, P1 is finally moved to O2 by M3. In the production of P1, R1 and M1, or R1, M2, R2, and M3 are used. Similarly, P2 is taken from I2 by M3, and after being processed by M3 it is moved from M3 to M2 by R2. Finally, after being processed by M2, P2 is moved to O3 by M2. To produce part type P2, M3, R2, and M2 are used. The S^3PR N model of the AMS is in Figure 6. Places p_7 and p_2 represent the operations of R1 and M1, respectively, i.e. one sequence at producing of the part type P1. Similarly, p_7 and p_3, p_5, p_6 represent the operations of R1, M2, R2, and M3, respectively, i.e. another sequence at producing of the part type P1. For production sequence of part type P2 p_8, p_4, p_9 represent the operations of M3, R2, and M2, respectively. The number of tokens in p_1, i.e., $M(p_1) = 5$, represents the number of concurrent activities that can take place for P1. The number of tokens in p_{10}, i.e., $M(p_{10}) = 3$, represents the number of concurrent activities that can take place for P2. Places p_{11} and p_{15} denote the resources R1 and M1, respectively. Places p_{12} - p_{15} denote shared resources M2, R2, and M3, respectively.

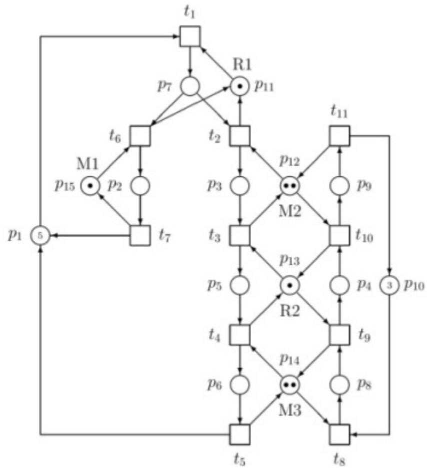

Figure 6
The S^3PR PN model of the AMS

Initial markings of p_{11}, p_{13}, and p_{15}, are all one as robots can hold one part and M1 can process one part at a time. Initial markings of p_{12} and p_{14} are two as either of M2 and M3 can process two parts at a time.

The net has 10 minimal siphons. However, seven of them are equal to (i.e. contain) traps and ergo they are prevented from emptying. As it was pointed out in [6], [7], and also mentioned above, such siphons cannot be emptied. Namely, only siphons being inclined to be emptied are dangerous, because they may lead to deadlocks. It means that there are such 3 strict minimal siphons (SMS) here. Namely,

$S_1 = \{p_4, p_6, p_{13}, p_{14}\}$; $S_2 = \{p_5, p_9, p_{12}, p_{13}\}$; $S_3 = \{p_6, p_9, p_{12}, p_{13}, p_{14}\}$

or in the matrix form

$$[SMS] = [\lambda]^T = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

The incidence matrix of N displayed in Figure 6 is $[N] = [Post]^T - [Pre]$, where

$$[Pre] = \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&1&0&0&0&0\\ 0&0&1&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&1&0\\ 0&0&0&1&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0&0&0&0\\ 0&1&0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0&1&0&0&0\\ 1&0&0&0&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0&0&1&0\\ 0&0&1&0&0&0&0&0&1&0&0\\ 0&0&0&1&0&0&0&1&0&0&0\\ 0&0&0&0&0&1&0&0&0&0&0 \end{pmatrix}$$

$$[Post]^T = \begin{pmatrix} 0&0&0&0&1&0&1&0&0&0&0\\ 0&0&0&0&0&1&0&0&0&0&0\\ 0&1&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&1&0&0\\ 0&0&1&0&0&0&0&0&0&0&0\\ 0&0&0&1&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&0&0&0&1\\ 0&1&0&0&0&1&0&0&0&0&0\\ 0&0&1&0&0&0&0&0&0&0&1\\ 0&0&0&1&0&0&0&0&0&1&0\\ 0&0&0&0&1&0&0&0&1&0&0\\ 0&0&0&0&0&0&1&0&0&0&0 \end{pmatrix}$$

$$[\eta]^T = [\lambda]^T.[N] = \begin{pmatrix} 0&0&-1&1&0&0&0&-1&1&0&0\\ 0&-1&1&0&0&0&0&0&-1&1&0\\ 0&-1&0&1&0&0&0&-1&0&1&0 \end{pmatrix}$$

The *rank* $([\eta]) = 2$, because only two of its rows are linearly independent. Namely, the third row is the sum of first and second one: $\eta_{S3} = \eta_{S1} + \eta_{S2}$. It means that there are two elementary siphons S_1, S_2 and one dependent siphon S_3. To ensure the deadlock prevention in our S³PR model, we have to add two control places V_{S1} and V_{S2} in order to control S_1 and S_2, respectively. The strongly dependent siphon S_3 can never be emptied in our case. Based on η_{S1} and η_{S2} we have the supervisor with the structure $[N]_S = [Post]_S^T - [Pre]_S$ resulting from $[\eta]$, where negative entries yield $[Pre]_S$, while positive entries yield $[Post]_S^T$. Namely,

$$[Pre]_S = \begin{pmatrix} 0&0&1&0&0&0&0&1&0&0&0\\ 0&1&0&0&0&0&0&0&1&0&0 \end{pmatrix}$$

$$[Post]_S^T = \begin{pmatrix} 0&0&0&1&0&0&0&0&1&0&0\\ 0&0&1&0&0&0&0&0&0&1&0 \end{pmatrix}$$

The first row of $[Pre]_S$ symbolizes directed arcs from V_{S1} to transitions of uncontrolled N displayed in Figure 6, while the first row of $[Post]_S^T$ symbolizes directed arcs from transitions of N to V_{S1}. Analogically, the second row of $[Pre]_S$ symbolizes directed arcs from V_{S2} to transitions of N, while the second row of $[Post]_S^T$ symbolizes directed arcs from transitions of N to V_{S2}. Hence, we obtain the new net N_1 displayed in Figure 7. It is composed of both the original uncontrolled net N displayed in Figure 6 and the supervisor N_S. Its incidence matrix $[N_1] = \begin{bmatrix}[N]\\ [N]_S\end{bmatrix}$.

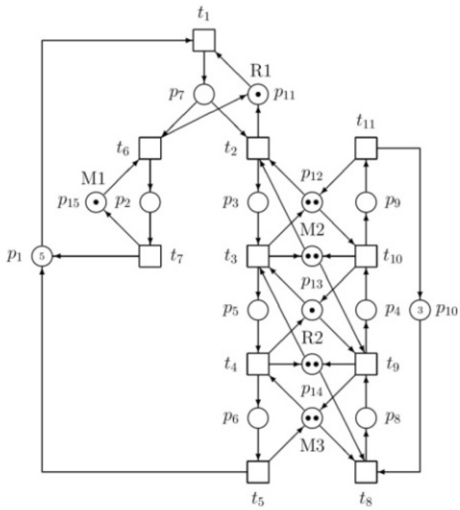

Figure 7
The controlled S³PR PN model of the AMS

3.1.2 Setting the Marking of Monitors

Now it is important to find a suitable marking of the control places (monitors) V_{S1} and V_{S2}. Namely, an inadequate setting of marking of these monitors may cause other deadlocks in the controlled plant.

For setting the marking of monitors V_{S_i}, i =1, n_m (n_m is a number of monitors) are valid the following general rules. Let $S = \{p_i, p_j, ..., p_k\}$ be SMS of a net system (N_0, M_0), where $N_0 = (P_0, T_0, F_0)$. Add a control place V_S to N_0 to make P-vector $I = (0, \ldots, 1_i, \ldots, 1_j, \ldots, 1_k, \ldots, 0, -1)^T$ be a P-invariant of a new net system (N_1, M_1), where $\forall p \in P_0 \backslash S$, $I(p) = 0$, $I(V_S) = -1$, $\forall p \in P_0$, $M_1(p) = M_0(p)$, and $[N_1]=[[N_0]^T | L_{VS}^T]^T$, where L_{V_S} is a row vector due to the addition of the place V_S. Let $M_1(V_S) = M_0(S) - \xi_S$, where $1 \leq \xi_S \leq M_0(S)$. Then, S is an invariant-controlled SMS and hence always marked at any reachable marking of the net system (N_1, M_1). Namely, I is a P-invariant and $\forall p \in (P_0 \cup \{V_S\}| \backslash S$, $I(p) < 0$. Note than $I^T.M_1 = I^T.M_0 = M_0(S) - M_1(V_S) = \xi_S > 0$. Thus, S is an *invariant-controlled* siphon.

To make a siphon S be always marked in a net system, we have to keep at least one token staying at S at any reachable marking of the net system. Suppose someway is found which controls S never to be emptied and the least number of tokens staying at S is denoted as, say, ξ. As mentioned above, ξ is called the *siphon control depth* variable. It is obvious the larger ξ is, the more behavior of the modeled system will be restricted, which, in Petri net formalism, means more reachable states will be forbidden. Therefore, let the siphon control depth variable be as small as possible, i.e. 1, whenever possible.

After N_0 is extended by V_S, the incidence matrix $[N_0]$ is extended by one row, denoted by L_{Vs}. Note that $I^T = (\lambda_S{}^T\text{-}1)$ and I is a P-invariant of N_1. Consequently, we have $I^T.[N_1] = 0^T$ and $\lambda_S{}^T.[N_0] - L_{Vs} = 0^T$. It means that $\lambda_S{}^T.[N_0] = L_{Vs}$ and $[N_1] = =[[N_0]^T \mid (\lambda_S{}^T.[N_0])^T]^T = [[N_0]^T \mid \eta_S]^T$. We can see that $L_{Vs} = \eta_S{}^T$.

It is easy to check from Figure 7 that $I_1 = p_4 + p_6 + p_{13} + p_{14} - V_{S_1}$ and $I_2 = p_5 + p_9 + p_{12} + p_{13} - V_{S2}$ are P-invariants of N_1. Clearly, $S_1 = \{p_4, p_6, p_{13}, p_{14}\}$ is invariant-controlled by I_1, since $\| I_1 \|^+ = S_1$ and $I_1{}^T.M_1 = M_1(S_1) - M_1(V_{S_1}) = 3 - 2 > 0$, and $S_2 = \{p_5, p_9, p_{12}, p_{13}\}$ is invariant-controlled by I_2, since $\| I_2 \|^+ = S_2$ and $I_2{}^T.M_2 = M_2(S_2) - M_2(V_{S_2}) = 3 - 2 > 0$. Here, $\| I_i \|^+$, $i = 1, 2$, are the positive supports of I_i.

S_3 is a redundant siphon. We can see that in uncontrolled net N the summary marking of S_3 (the sum of marking of its places) is $M_0(S_3) = 5$ while the summary marking of S_1 and S_2 are $M_0(S_1) = M_0(S_2) = 3$. Here, $M_0(S_i)$ means the marking of S_i.

Let $\xi_{S1} = \xi_{S2} = 1$, we have $M_0(S_3) > M_0(S_1) + M_0(S_2) - \xi_{S1} - \xi_{S2}$. Thus, S_3 can never be emptied after S_1 and S_2 are controlled via adding two control places (monitors) V_{S_1}, V_{S_2}, as it is shown in Figure 7.

It is easy to check that S_1 and S_2 can never be emptied. In such a way deadlocks in our S³PR net are prevented and controlled system can operate safely (without deadlocks) and reliably.

3.2 Method Preventing Strict Minimal Siphons from Emptying

This method, sometimes called as *classical method*, consists of the work with circuits, the set of holders of resources, and with complementary siphons.

3.2.1 Circuits and Complementary Siphons in S³PR

Let $\boldsymbol{C}$ be a circuit of N and x and y be two nodes of $\boldsymbol{C}$. Node x is said to be *previous* to y *iff* (if and only if) there exists a path in $\boldsymbol{C}$ from x to y, the length of which is greater than one and does not pass over the idle process place p^0. This fact is denoted by $x \prec_C y$. In general, the symbol $\prec$ means a *generic strict order relation*, while $\nprec$ symbolizes the assertion *'does not precede'*.

Let x and y be two nodes in N. Node x is said to be previous to y in N *iff* there exists a circuit $\boldsymbol{C}$ such that $x \prec_C y$. This fact is denoted by $x \prec_N y$.

Let x be a node and $A \subseteq P \cup T$ be a set of nodes in N. Then $x \prec_N A$ *iff* there exists a node $y \in A$ such that $x \prec_N y$ and $A \prec_N x$ *iff* there exists a node $y \in A$ such that $y \prec_N x$.

For $r \in P_R$, $H(r) = {}^{\bullet\bullet}r \cap P_A$ (the operation places P_A that use r), is called the set of holders of r.

$[S] = (\bigcup_{r \in S_R} H(r)) \setminus S$ is called the *complementary set* of the siphon S.

In the net N in Figure 6, $\boldsymbol{C} = p_1t_1p_7t_2p_3t_3p_5t_4p_6t_5p_1$ is a circuit and the *elementary path* $EP(p_7, p_6) = p_7t_2p_3t_3p_5t_4p_6$ is a path in $\boldsymbol{C}$. The support of $EP(p_7, p_6)$ is $\{p_7, t_2, p_3, t_3, p_5, t_4, p_6\}$ and the support of $\boldsymbol{C}$ is $\{p_1, t_1, p_7, t_2, p_3, t_3, p_5, t_4, p_6, t_5\}$. Clearly, we have $p_7 \prec_C p_6$ and $p_7 \prec_N p_6$.

Consider the same AMS with the same structure like that in Figure 4 - Figure 6, however, with another initial marking of resources, displayed in Figure 8 (left).

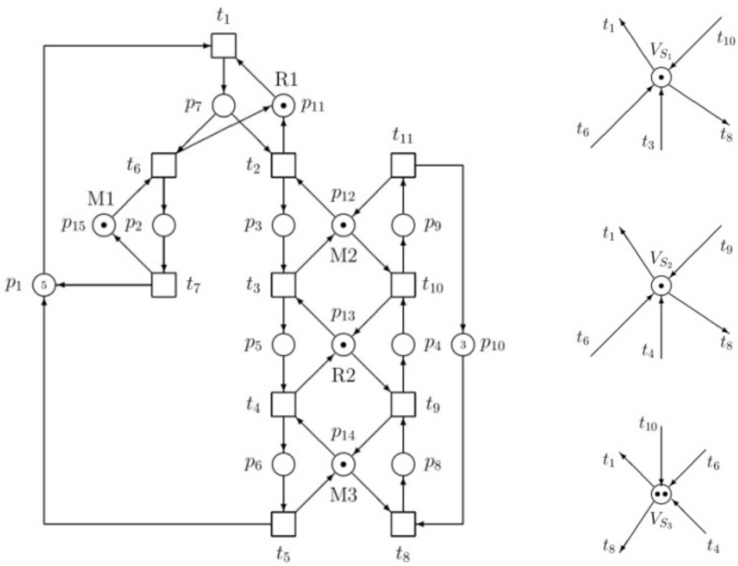

Figure 8

The uncontrolled S[3]PR PN model of the AMS (left) and thee monitors creating the controller (right)

Circuits are structural parameters. Therefore, $\boldsymbol{C}$ introduced above in connection with Figure 6 it is valid here too. There are 3 minimal siphons in the net introduced in Figure 8, namely, $S_1 = \{p_5, p_9, p_{12}, p_{13}\}$, $S_2 = \{p_4, p_6, p_{13}, p_{14}\}$, and $S_3 = \{p_6, p_9, p_{12}, p_{13}, p_{14}\}$. They are the same as those in connection with Figure 6. Their complementary siphons are $[S_1] = \{p_3, p_4\}$, $[S_2] = \{p_5, p_8\}$, and $[S_3] = \{p_3, p_4, p_5, p_8\}$.

3.2.2 Downstream and Upstream Siphons in S^3PR

Let $\Delta^+(t)$ ($\Delta^-(t)$) denote the set of downstream (upstream) siphons of a transition t and $\boldsymbol{P}_S$ denote the adjoint set of a siphon S in an S^3PR $N = \bigcirc_{i=1}^{n} N_i = (P^0 \cup P_A \cup P_R, T, F)$. Then

1. $\Delta^+ : T \longrightarrow 2^{\Pi}$ (2^{Π} is the power set of the set Π being the set of minimal siphons) is a mapping defined as follows: If $t \in T_i$, then $\Delta^+(t) = \{S \in \Pi \mid t \prec_{\overline{N_i}} [S]^i\}$. If $S \in \Delta^+(t)$ then the set $[S]^i$ is *reachable* from t, i.e., there exists a path in $\overline{N_i}$ leading from t to an operation place $p \in P_{Ai}$ that is *not included* in S but uses a resource of S, where $[S]=\bigcup_{i=1}^{n} [S]^i$, $P_A = \bigcup_{i=1}^{n} P_{Ai}$, and $[S]^i = [S] \cap P_{Ai}$.
2. 2. $\Delta^- : T \longrightarrow 2^{\Pi}$ is a mapping defined as follows: If $t \in T_i$, then $\Delta^-(t) = \{S \in \Pi \mid [S]^i \prec_{\overline{N_i}} t\}$.
3. $\forall i \in N_n$, $\forall S \in \Pi$, $\boldsymbol{P}_S^i = [S]^i \cup \{p \in P_{Ai} \mid p \prec_{\overline{N_i}} [S]^i\}$, and $\boldsymbol{P}_S = \bigcup_{i=1}^{n} \boldsymbol{P}_S^i$, where $N_n = \{1, 2, \ldots, n\}$.

The downstream siphons are $\Delta^+(t_1) = \Delta^+(t_2) = \Delta^+(t_8) = \{S_1, S_2, S_3\}$, $\Delta^+(t_3) = \{S_2, S_3\}$, and $\Delta^+(t_4) = \Delta^+(t_{10}) = \emptyset$. Analogically, upstream siphons are $\Delta^-(t_1) = \Delta^-(t_2) = \Delta^-(t_6) = \Delta^-(t_7) = \emptyset$, $\Delta^-(t_3) = \{S_1\}$, $\Delta^-(t_4) = \Delta^-(t_5) = \{S_1, S_2, S_3\}$. The adjoint sets are $\boldsymbol{P}_{S1} = \boldsymbol{P}_{S1}^1 \cup \boldsymbol{P}_{S1}^2 = (\{p_3\} \cup \{p_7\} \cup \{p_4\} \cup \{p_8\} = \{p_3, p_4, p_7, p_8\}$, $\boldsymbol{P}_{S2} = \boldsymbol{P}_{S2}^1 \cup \boldsymbol{P}_{S2}^2 = (\{p_5\} \cup \{p_7, p_3\} \cup \{p_8\} = \{p_3, p_5, p_7, p_8\}$, and $\boldsymbol{P}_{S3} = \boldsymbol{P}_{S3}^1 \cup \boldsymbol{P}_{S3}^2 = (\{p_3, p_5\} \cup \{p_7\} \cup \{p_4, p_8\} = \{p_3, p_4, p_5, p_7, p_8\}$.

3.2.3 Implementation of Monitors and Setting their Markings

The net $(N_V, M_{0V}) = (P_A \cup P^0 \cup P_R \cup P_V, T, F \cup F_V, M_{0V})$ is the controlled system of the net (N, M_0) *iff*:

1. $P_V = \{V_S \mid S \in \Pi\}$ is a set of monitors and there is a bijective mapping between Π and P_V (i.e. one-to-one and onto mapping; it can be inverted).
2. $F_V = F_V^1 \cup F_V^2 \cup F_V^3$ with
   $F_V^1 = \{(V_S, t) \mid S \in \Delta^+(t), t \in P^{0\bullet}\}$
   $F_V^2 = \{(t, V_S) \mid t \in [S]^\bullet, S \notin \Delta^+(t)\}$
   $F_V^3 = \bigcup_{i=1}^{n} \{(t, V_S) \mid t \in T_i \setminus P^{0\bullet}, S \notin \Delta^-(t), {}^\bullet t \cap P_{Ai} \subseteq \boldsymbol{P}_S^i, t \nprec [S]^i\}$
3. M_{0V} is defined as follows:
   - $\forall p \in P_A \cup P^0 \cup P_R$, $M_{0V}(p)$; and
   - $\forall V_S \in P_V$, $M_{0V}(V_S) = M_0(S) - 1$.

In our example three monitors are needed to prevent three SMS from being emptied.

Take first the siphon $S_1 = \{p_5, p_9, p_{12}, p_{13}\}$ as an example. We can see that $P^0 = \{p_1, p_{10}\}$. Thus, $P^{0\bullet} = \{t_1, t_8\}$. As a result, we have $\{(V_{S1}, t_1), (V_{S1}, t_8)\} \subseteq F_V^1$.

Due to $[S_1] = \{p_3, p_4\}$, $[S_1]^\bullet = \{t_3, t_{10}\}$. We can see that $S_1 \notin \Delta^+(t_3)$ and $S_1 \notin \Delta^+(t_{10})$. Consequently, $\{(t_3, V_{S1}), (t_{10}, V_{S1})\} \subseteq F_V^2$. Let us find the arcs related to V_{S1} in F_V^3. Put $T_\alpha = (T_1 \setminus P^{0\bullet}) \cup (T_2 \setminus P^{0\bullet})$; $T_\beta = \{t |\ S_1 \notin \Delta^-(t), t \in T\}$; $T_\gamma = \{t |\ {}^\bullet t \cap P_{A1} \subseteq \boldsymbol{P}_S{}^1\} \cup \{t |\ {}^\bullet t \cap P_{A2} \subseteq \boldsymbol{P}_S{}^2\}$; $T_\delta = \{t |\ t \nprec [S_1]^1\} \cup \{t |\ t \nprec [S_1]^2\}$. Hence, $T_\alpha = \{t_2, t_3, t_4, t_5, t_6, t_7, t_9, t_{10}, t_{11}\}$; $T_\beta = \{t_1, t_2, t_6, t_7, t_8, t_9\}$; $T_\gamma = \{t_2, t_3, t_6, t_9, t_{10}\}$; $T_\delta = \{t_3, t_4, t_5, t_6, t_7, t_{10}, t_{11}\}$. It can be seen that $T_\alpha \cap T_\beta \cap T_\gamma \cap T_\delta = \{t_6\}$. Thus, $(t_6, V_{S1}) \in F_V^3$. For siphons S_2, S_3, monitors V_{S2}, V_{S3} can be added where $\{(V_{S2}, t_1), (V_{S2}, t_8), (V_{S3}, t_1), (V_{S3}, t_8)\} \subseteq F_V^1$, $\{(t_4, V_{S2}), (t_9, V_{S2}), (t_4, V_{S3}), (t_{10}, V_{S3})\} \subseteq F_V^2$, and $\{t_6, V_{S2}), (t_6, V_{S3})\} \subseteq F_V^3$. The supervised system is displayed in Figure 8.

As to marking of monitors $M_0(V_{S1}) = M_0(p_4) + M_0(p_9) + M_0(p_{12}) + M_0(p_{13}) - 1 = 0 + 0 + 1 + 1 - 1 = 2 - 1 = 1$; $M_0(V_{S2}) = M_0(p_4) + M_0(p_9) + M_0(p_{12}) + M_0(p_{13}) - 1 = 0 + 0 + 1 + 1 - 1 = 2 - 1 = 1$; $M_0(V_{S3}) = M_0(p_6) + M_0(p_9) + M_0(p_{12}) + M_0(p_{13}) + M_0(p_{14}) - 1 = 0 + 0 + 1 + 1 + 1 - 1 = 3 - 1 = 2$.

Three monitors in Figure 8 (right) controlling the plant with the PN model (left) are drawn separately in order to avoid confusing at drawing crisscross mutual inter-connections between the PN model of uncontrolled plant and monitors. In spite of the separate drawing, it is clear which monitors are connected with which transitions.

Conclusions

Process of dealing with the allocation of resources may complicate prevention of S^3PR net systems from deadlocks. Two approaches to deadlock prevention of S^3PR net systems modeling automated manufactory systems (AMS) containing common resources (e.g. competitively utilized several manufacturing devices or other kinds of resources, like buffers of parts, etc.) were presented in this paper.

First of approaches is based on elementary siphons of the PN models of S^3PR net systems, while the second one is based on preventing strict minimal siphons of such PN models from being emptied, in another way. The former approach is more analytical (better expressed in analytical terms) and more friendly for processing by computer because it uses linear algebra. The latter approach, based on preventing strict minimal siphons from being emptied, utilizes the analysis of circuits, computing the holders of resources and complementary siphons, what needs some preprocessing or more complicated algorithm able to handle operations from the set theory. On the other hand, also in the former approach the situation may be a little hindered by the needfulness to deal with strongly or weakly dependent siphons if it is necessary (i.e. if they are not automatically prevented before emptying already within the framework of preventing the elementary siphons).

In any case, both approaches are very useful in the deadlock prevention in AMS with the requirement of correct resource allocation. Moreover, they are very useful not only for reliable control of existing AMS, but also at the design of new deadlock-free AMS like those.

For computing S^3PR siphons and traps themselves, the MATLAB based tool GPenSIM [24] was used. MATLAB itself (or at least GNU Octave) is suitable also for computer application both of the deadlock prevention methods.

Benefits following from the application of such deadlock prevention methods yield deadlock free RAS designed off-line (still before their actual deployment in practice). It means that such methods intensively help us at the AMS design. This at the least rapidly decreases a risk of defects in operation of real AMS as well as prevents their shutdowns. In such a way, it is possible to avoid significant economic losses. It is main advantage of the approaches presented in this paper. However, other external disturbances unrelated to deadlocks, cannot be prevented in such a way.

Acknowledgement

This work was partially supported by the Slovak Grant Agency for Science VEGA under Grant No. 2/0020/21.

References

[1] C. A. Petri: Communication with Automata, Ph. D. Thesis. Technical University od Darmstadt, Germany, 1962, 128 pages (in German)

[2] W. Reisig: Petri Nets, Springer, Berlin Heidelberg, 1985

[3] T. Murata: Petri Nets: Properties, Analysis and Applications, Proceedings of the IEEE, Vol. 77, No. 4, 1989, pp. 541-580

[4] J. Desel, W. Reisig: Place/Transition Petri Nets. In: W. Reisig, G. Rozenberg (Eds.): Advances of Petri Nets, Lecture Notes in Computer Science, Vol. 1941, Springer, Heidelberg, 1998, pp. 122-173

[5] F. Čapkovič: Modeling and Control of Discrete-Event Systems with Partial Non-Determinism Using Petri Nets. Acta Polytechnica Hungarica, Vol. 17, No. 4, 2020, pp. 47-66

[6] F. Čapkovič: Control of Deadlocked Discrete-Event Systems Using Petri Nets. Acta Polytechnica Hungarica, Vol. 19, No. 2, 2022, pp. 213-233

[7] F. Čapkovič: Modeling and Control of Resource Allocation Systems within Discrete-Event Systems by Means of Petri Nets - Part 1: Invariants, Siphons and Traps in Deadlock Avoidance. Computing and Informatics, Vol. 40, No. 3, 2021, pp. 648-689

[8] X. Guan, Y. Li, J. Xu, C. Wang, S. Wang: A Literature Review of Deadlock Prevention Policy Based on Petri Nets forAutomated Manifacturing Systems, International Journal of Digital Content Technology and its Applications (JDTCA), Vol. 6, 2012, pp. 426-433

[9] H. Yue, K. Y. Xing, H. S. Hu,W. Wu, H. Y. Su: Petri-Net-Based Robust Supervisory Control of Automated Manufacturing Systems, Control Engineering Practice, Vol. 54, 2016, pp. 176-189

[10] A. Farooq, H. Huang, X. L. Wang: Petri Net Modeling and Deadlock Analysis of Parallel Manufacturing Processes with Shared-Resources, The Journal of Systems and Software, Vol. 83, 2010, pp. 675-688

[11] H. Hu, Y. Liu, L. Yuan: Supervisor Simplification in FMs: Comparative Studies and New Results Using Petri Nets, IEEE Transactions on Control System Technology, Vol. 24, 2016, pp. 81-95

[12] Z. W. Li and M. C. Zhou: Elementary Siphons of Petri Nets and their Application to Deadlock Prevention in Flexible Manufacturing Systems. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, Vol. 34, No. 1, 2004, pp. 38-51, doi:10.1109/TSMCA.2003.820576

[13] Z. W. Li and M. C. Zhou: Deadlock Resolution in Automated Manufacturing Systems. A Novel Petri Net Approach, London, Springer Press, 2009

[14] G. Y. Liu: Supervisor Synthesis for Automated Manufacturing Systems Based on Structure Theory of Petri Nets. Doctoral Thesis, Paris Graduate School for Informatics, Telecommunications and Electronics, Paris, France, 2014

[15] E. G. Coffman, M. J. Elphick, A. Shoshani: Systems Deadlocks, ACM Computing Surveys, Vol. 3, No. 2, 1971, pp. 66-78

[16] G. Y. Liu and K. Barkaoui: A Survey of Siphons in Petri Nets. Information Sciences, Vol. 363, 2016, pp. 198-220. Available at: https://doi.org/10.1016/j.ins.2015.08.037

[17] C. F. Zhong and Z. W. Li: A deadlock prevention approach for flexible manufacturing systems without complete siphon enumeration of their Petri net models. Engineering with Computers, Vol. 25, 2009, pp. 269-278

[18] D. Y. Chao and Y. L. Pan: Uniform Formulas for Compound Siphons, Complementary Siphons and Characteristic Vectors in Deadlock Prevention of Flexible Manufacturing Systems. Journal of Intelligent Manufacturing, Vol. 26, No. 1, 2015, pp. 13-23, doi: 10.1007/s10845-013-0757-7

[19] Z. W. Li and M. C. Zhou: Control of Elementary and Dependent Siphons in Petri Nets and Their Application. IEEE Transactions on Systems, Man, Cybernetics, Part A, Systems and Humans, Vol. 38, No. 1, 2008, pp. 133-148, doi: 10.1109/TSMCA.2007.909548

[20] J. C. Luo, Z. Q. Liu and M. C. Zhou: A Petri Net Based Deadlock Avoidance Policy for Flexible Manufacturing Systems with Assembly Operations and Multiple Resource Acquisition. IEEE Transactions on Industrial Informatics, Vol. 15, No. 6, 2019, pp. 3379-3387

[21] E. A. Nasr, A. M. El-Tamimi, A. Al-Ahmari, H. Kaid: Comparison and Evaluation of Deadlock Prevention Methods for Different Size Automated Manufacturing Systems. Hindawi Publishing Corporation, Mathematical Problems in Engineering, Vol. 2015, Article ID 537893, 19 pages. http://dx.doi.org/10.1155/2015/53789

[22] D. Y. Chao: Improved Controllability Test for Dependent Siphons in S^3PR Based on Elementary Siphons. Asian Journal of Control, Vol. 12, No. 3, 2010, pp. 377-391

[23] M. P. Cabasino, A. Giua and C. Seatzu: Modeling and supervisory control of railway networks using Petri nets, IEEE Transactions on Automation Science and Engineering, Vol. 5, No. 3, 2008, pp. 431-445

[24] R. Davidjaruh: GPenSIM, General Purpose Petri Net Sumulator for MATLAB Platform. Available at: http://www.davidrajuh.net/gpensim/