Computing and Informatics, Vol. 37, 2018, 1258–1292, doi: 10.4149/cai_2018_5_1258

PETRI NETS AT MODELLING AND CONTROL OF DISCRETE-EVENT SYSTEMS CONTAINING NONDETERMINISM – PART 1

František Čapkovič

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

Abstract. Discrete-Event Systems are discrete in nature, driven by discrete events. Petri Nets are one of the mostly used tools for their modelling and control synthesis. Place/Transitions Petri Nets, Timed Petri Nets, Controlled Petri Nets are suitable when a modelled object is deterministic. When the system model contains uncontrollable/unobservable transitions and unobservable/unmeasurable places or other failures, such kinds of Petri Nets are insufficient for the purpose. In such a case Labelled Petri Nets and/or Interpreted Petri Nets have to be used. Particularities and mutual differences of individual kinds of Petri Nets are pointed out and their applicability to modelling and control of Discrete-Event Systems are described and tested.

Keywords: Analysing, control synthesis, controlled Petri nets, discrete-event systems, interpreted Petri nets, modelling, labelled Petri nets, place/transition Petri nets, timed Petri nets, uncertainty, unobservable/uncontrollable transitions, unmeasurable/unobservable places

Mathematics Subject Classification 2010: 93-C65, 93-C30

1 INTRODUCTION AND PRELIMINARIES

Discrete-Event Systems (DES) are frequently modelled by Petri Nets (PN). As to their structure, PN are [9] bipartite directed graphs with two kinds of nodes –

places $p_i \in P$, $i = 1, \dots, n$ (i.e. $|P| = n$) and transitions $t_j \in T$, $j = 1, \dots, m$ (i.e. $|T| = m$) – and two kinds of arcs – the arcs directed from places to transitions $F \subseteq P \times T$ and the arcs directed from transitions to places $G \subseteq T \times P$. Here, $P \cap T = \varnothing$ and $P \cup T \neq \varnothing$, where $\varnothing$ symbolizes an empty set. Hence, $B = F \subseteq P \times T \cup G \subseteq T \times P$ represents the PN structure. Thus, the net structure is [29, 30] $N = \langle P, T, B \rangle$. The preset of a transition t (i.e. the set of its input places) is defined as ${}^{(p)}t = \{p|(p,t) \in B\}$, while the postset of t (i.e. the set of its output places) is defined as $t^{(p)} = \{p|(t,p) \in B\}$. On the other hand, the preset of a place p (i.e. the set of its input transitions) is defined as ${}^{(t)}p = \{t|(t,p) \in B\}$ while the postset of p (i.e. the set of its output transitions) is defined as $p^{(t)} = \{t|(p,t) \in B\}$. If for $p \in P$, $t \in T$, $\{(p,t) \in B) \Rightarrow (t,p) \notin B\}$, i.e., if no self-loops occur in PN, then the net is said to be *pure*. A transition is said to be the source transition if ${}^{(p)}t = \varnothing$ and the sink transition if $t^{(p)} = \varnothing$.

Places model particular operations or activities of DES, states of which are expressed by marking – i.e. by the number of tokens $n_t \in \mathbb{Z}_{\geq 0}$ put into them. It means that marking m is a vector $m : P \to \mathbb{Z}_{\geq 0}$ where $\mathbb{Z}_{\geq 0}$ represents positive integers including 0. PN transitions model the discrete events in DES. A transition can be disabled (when it cannot be fired) or enabled (when it can be fired). Of course, the enabled transition might be, but need not to be, fired. An event modelling a failure is fired spontaneously. A set of transitions $\mathcal{T} \subseteq T$ is enabled by means of the marking m if $\forall p \in P$, $m(p) \geq |p^{(t)} \cap \mathcal{T}|$. It means that $m(p)$ is greater than the number of transitions in $\mathcal{T}$ for which p is the input place or equal to this number. The occurrence of a discrete event is modelled by means of firing the corresponding enabled transition.

Theoretically, more than one transition can be fired at any instant [29, 62]. Thus two possibilities offer:

1. to fire more than one transition at any instant – so called concurrency assumption (in short the C assumption);
2. to fire only one of the transitions at any instant – so called no concurrency assumption (in short the NC assumption).

In the former case, if a set of transitions $\mathcal{T} \subseteq T$ is enabled at marking m, then $\mathcal{T}$ may fire and the new marking m' is obtained as $m'(p) = m(p) + |{}^{(t)}p \cap \mathcal{T}| - |p^{(t)} \cap \mathcal{T}|$. It means that firing the set of transitions $\mathcal{T} \subseteq T$ causes that one token will be removed from each place $p \in {}^{(p)}t$ and one token will be added to each place $p \in t^{(p)}$. In literature about PN, the case C is used very rarely.

The latter case NC is usual in the total most of PN literature. Here, it is assumed that only a single transition is fired at any instant. Under such assumption, $\mathcal{T}$ is a singleton set (a set having exactly one element). Here, in this paper, the NC assumption will be applied.

A firing sequence from an initial marking m_0 is a sequence of transition sets $\mathcal{U} = \{\tau_1 \tau_2 \dots \tau_k\}$ such that $m_0[\tau_1 > m_1[\tau_2 > \dots m_{k-1}[\tau_k > m_k$. The set may also be empty. The notation $m_0[\mathcal{U}$ denotes that the sequence $\mathcal{U}$ can be fired at

m_0 and the notation $m_0[\mathcal{U} > m_k$ denotes that the firing of $\mathcal{U}$ yields m_k. Under the NC assumption, each τ_i is a singleton set, and $\mathcal{U}$ is a sequence of transitions. To denote that by firing of $\mathcal{U}$ the state m_k can be reached from m_0 it can be written that $m_0[\mathcal{U} > m_k$. This is connected with the reachability of PN markings (states). In general, marking m is reachable in a net system $\langle N, m_0\rangle$ if there exists a firing sequence $\mathcal{U}$ such that $m_0[\mathcal{U} > m$. When the net system $\langle N, m_0\rangle$ is given, the set of reachable markings is $R(N, m_0)$. Markings reachable from a given initial marking can be expressed by means of the reachability tree (RT) and/or the reachability graph (RG). RG arises from RT by connecting all RT nodes with the same name into one node. The incidence matrix of RG is the same as that of RT. A certain RT appertains to each initial state of PN unambiguously. Unfortunately, the opposite relation does not exist. In general, it is practically impossible to unambiguously obtain PN from a given RT.

In some PN the number of states (i.e. the reachability set) is also infinite. Consequently, RT (RG) is also infinite. To compute a substitutional finite graph, so called coverability graph (CG), a different algorithm has to be used. Each arc corresponds to a transition, but each node corresponds either to a single reachable marking or it represents an infinite set of reachable markings – in such case loops are originating in corresponding CG nodes.

PN-based models can be created either intuitively, based on a creator's empirical experience and knowledge acquired from the external observation of the behaviour of real systems (a usual approach) or by means of a systematic approach – see e.g. [6, 60, 15, 16, 17, 18, 3, 24]. However, during the model creation PN properties should be considered. The basic properties (as safeness, liveness, boundedness, reachability, reversibility, deadlock-freeness, conservativeness, etc.) are defined e.g. in basic literature [49, 47, 11] and in many other papers.

A net N is well-formed if there exists a marking m_0 of N such that $\langle N, m_0\rangle$ is a live and bounded system.

More details about PN can be found especially in the fundamental literature [49, 47, 11] but also in many other newer sources which are specialized in extending the fundamental knowledge towards different application areas (e.g. supervision, control, etc.).

1.1 Place/Transitions Petri Nets

The later naming of above mentioned PN is Place/Transition PN (P/T PN) – see e.g. [11]. Because linear algebra can be used at PN-based modelling DES, it is useful to apply the better arranged (from the mathematical point of view) vector notation [7, 8] at modelling DES by means of P/T PN. Because from the system point of view markings correspond with state vectors of places and the transitions correspond with control variables, the marking evolution (*dynamics*) of P/T PN can be expressed as the following restricted linear discrete state equation:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{B}.\mathbf{u}_k, \quad k = 0, 1, \ldots, N, \tag{1}$$

$$\mathbf{F}.\mathbf{u}_k \leq \mathbf{x}_k. \tag{2}$$

Here, $\mathbf{x}_k = (\sigma^k_{p_1}, \ldots, \sigma^k_{p_n})^T$ is the state vector of the places in the step k with $\sigma^k_{p_i} \in \{0, 1, \ldots, \infty\}$, $i = 1, \ldots, n$; $\mathbf{u}_k = (\gamma^k_{t_1}, \ldots, \gamma^k_{t_m})^T$ is the state vector of the transitions in the step k (named as the control vector) with $\gamma^k_{t_j} \in \{0, 1\}$, $j = 1, \ldots, m$, where 0 denotes the disabled transition and 1 denotes the enabled one; $\mathbf{B} = \mathbf{G}^T - \mathbf{F}$ is the structural matrix. Here, the matrices $\mathbf{B}$, $\mathbf{G}$, $\mathbf{F}$ correspond, respectively, to the sets B, G, F. $\mathbf{F} \in \mathbb{Z}^{n\times m}_{\geq 0}$, $\mathbf{G} \in \mathbb{Z}^{m\times n}_{\geq 0}$ and $\mathbf{B} \in \mathbb{Z}^{n\times m}$, where $\mathbb{Z}$ represents all integers including zero.

For illustration, an example of P/T PN is displayed in Figure 1. Its incidence matrices are

$$\mathbf{F} = \begin{pmatrix} 0&1&0&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&1&0&0\\ 0&0&0&1&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&1\\ 0&0&0&0&0&1&0&0&0&0\\ 0&0&1&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&1&0&0&0\\ 1&0&0&0&0&0&0&0&0&0 \end{pmatrix}; \quad \mathbf{G}^T = \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&1&0&0\\ 0&0&0&1&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&1\\ 0&0&0&0&0&1&0&0&0&0\\ 0&0&1&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&1&0&0&0 \end{pmatrix}. \tag{3}$$

The initial state $\mathbf{x}_0 = (0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1)^T$ displayed in Figure 1 yields RT given in Figure 2.

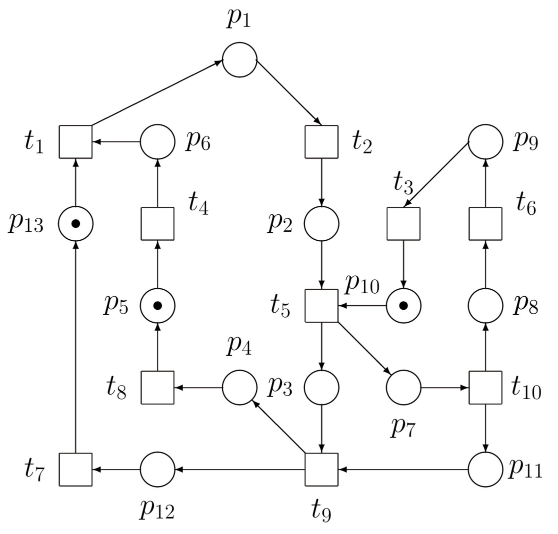

Figure 1. An example of P/T PN

The nodes of the corresponding RT/RG creating the state space of the PN are the vectors being the columns of the following reachability matrix:

$$\mathbf{X}_r = \begin{pmatrix}
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1\\
0&0&0&0&1&1&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&1&1&0&1&1&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&1&0&0&1&1&0&0&1&1&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1&1&1&1&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&1&0&0&1&1&0&0&0&1&1&0&0&0&0&1&0&0&1&0&1&0\\
0&0&0&0&0&0&1&0&0&1&0&0&0&1&1&0&0&0&0&1&1&0&0&1&0&1&0&1\\
1&1&1&1&0&0&0&0&1&0&0&0&1&0&0&0&0&1&1&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&1&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&1&0&1&1&0&1&0&1&0&1&0&1&0&1&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&0&1&0&0&1&0&1&0&1&0&1&0&1&0&1&0&0&0&0
\end{pmatrix} \quad (4)$$

Connecting the RT nodes with the same name, RT turns to RG. It expresses the possibilities of passing the system from a given state to another state by means of firing the transitions assigned to the corresponding RT arcs.

It is still necessary to mention the Parikh's vector. That is to say, the development of the system (1) is the following:

$$\begin{aligned}
\mathbf{x}_k &= \mathbf{x}_{k-1} + \mathbf{B}.\mathbf{u}_{k-1} = \mathbf{x}_{k-2} + \mathbf{B}.(\mathbf{u}_{k-1} + \mathbf{u}_{k-2}) = \ldots \\
&= \mathbf{x}_0 + \mathbf{B}.(\mathbf{u}_0 + \mathbf{u}_1 + \ldots + \mathbf{u}_{k-1}) = \mathbf{x}_0 + \mathbf{B}.\mathbf{v} \qquad (5)
\end{aligned}$$

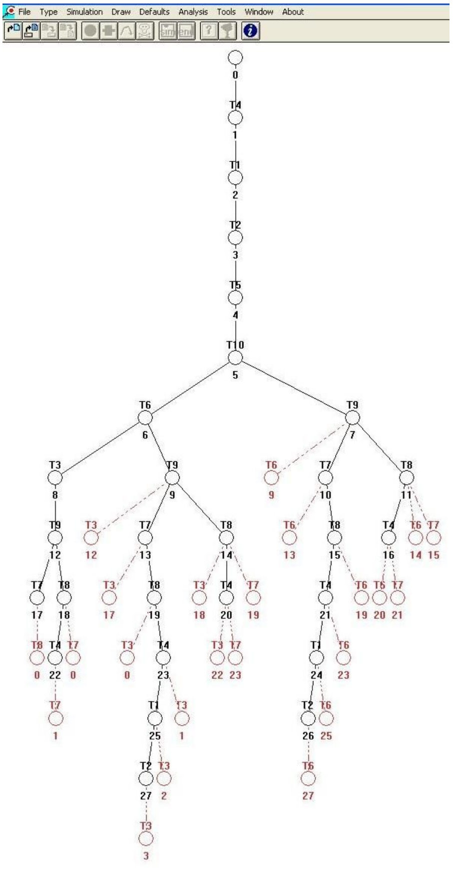

Figure 2. The RT of the P/T PN

where $\mathbf{v} = \mathbf{u}_0 + \mathbf{u}_1 + \ldots + \mathbf{u}_{k-1}$ is named as the Parikh's vector. It yields information on how many times the particular transitions are fired during the evolution of P/T PN from the initial state $\mathbf{x}_0$ to a prescribed terminal state $\mathbf{x}_k$.

There exist some special kinds of P/T PN. The mostly used are the following:

- P/T PN where every place has exactly one incoming arc, and exactly one outgoing arc are named as *marked graphs* (MG). Mathematically expressed $\forall p \in P : |p^{(t)}| = |^{(t)}p| = 1$. No conflicts in MG occur, but a concurrency can be expressed there. In a graph interpretation, MG is a graph where each place represents an arc and each transition represents a node.
- P/T PN where every transition has exactly one incoming arc, and exactly one outgoing arc and all markings have exactly one token then are named as *state machines* (SM). Mathematically expressed $\forall t \in T : |t^{(p)}| = |^{(p)}t| = 1$. Concur-

rency cannot be expressed there, but conflicts appear caused by more outgoing transitions from a place (i.e. a kind of uncertainty or nondeterminism).

- P/T PN where every arc from a place to a transition is either the only arc from that place, or the only arc to that transition, are named as a *free-choice PN* – see Figure 3.

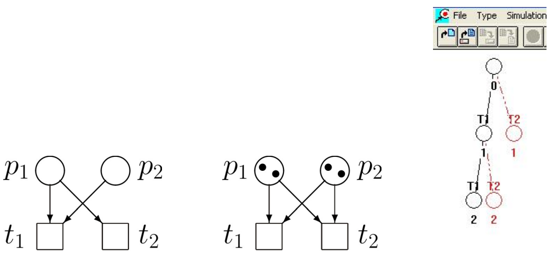

Figure 3. A fragment of the free-choice PN (left), a simple free-choice PN (in the centre) and its RT (right)

A net $N = \langle P, T, B \rangle$ is free-choice [12] if $(p, t) \in B$ implies ${}^{(p)}t \times p^{(t)} \subseteq B$ for every place p and every transition t. A net system $\langle N, M_0 \rangle$ is free-choice if its underlying net N is free-choice. In [12] also the fundamental property of a free-choice PN was proved. If a marking of N enables some transition of $p^{(t)}$, then it enables every transition of $p^{(t)}$.

1.2 Timed Petri Nets

Timed Petri Nets (TPN) [64, 59, 7, 9] are based on P/T PN structure. They rise by introducing time into P/T PN transitions through their duration function $D : \mathcal{T} \rightarrow \mathbb{Q}_0^+$ ($\mathbb{Q}_0^+$ symbolizes non-negative rational numbers). The timing can be deterministic (time delays) or nondeterministic (expressed by a probability distribution – e.g. exponential, discrete uniform, etc.). More details can be found in [59, 7, 9]. As to the DES control, at P/T PN a supervisor can be synthesized. By means of TPN the performance evaluation of the supervised system can be tested.

1.3 Closing Remark

The P/T PN and TPN are very useful tools for modelling, analysing and control of DES. Such models of DES represent an initial point (a base) for some procedures of DES control synthesis. However, in the course of time other kinds of PN arising from P/T PN were developed, that are more suitable for specific kinds of DES, especially designated for control – Controlled Petri Nets, and for control of DES containing nondeterminism in the form of unobservable/uncontrollable transitions and/or unmeasurable places – Labelled Petri Nets and Interpreted Petri Nets.

Because of the extent the problem (a broad issue), publication is divided into two separate parts – Part 1 and Part 2. The particular parts will be published as individual papers in successive steps. This paper represents the individual Part 1. It is devoted to presentation of the mentioned kinds of PN and marking the possibilities for applicability at modelling DES (in the first place) and also to point out that the PN can be used for control purposes. In the planned consecutive Part 2 more detailed case studies on applicability at control of different kinds of DES, will be introduced, including the error recovery approach at a fault occurrence. The Part 2 will be published as the separate paper.

2 SOME OF OTHER KINDS OF PN RELATED TO DES CONTROL

P/T PN is a good mathematical tool for modelling and analysing DES. However, creation of the model is not sufficient for the DES control. After creating the model of the system to be controlled, it is necessary to synthesize the procedure of control. The best way is when it is possible to utilize the model for the control synthesis not only in deterministic cases when all transitions are controllable and all places are measurable but also in cases where uncontrollable transitions and non-measurable places or even faults occur in the model. Then, two sets of transitions arise. That is to say, the set T_c of controllable transitions and the set T_u of uncontrollable transitions. $T = T_c \cup T_u$. Analogically, also two sets of places arise – the set P_m of the measurable places and the set P_{nm} of unmeasurable places, $P = P_m \cup P_{nm}$. Because of the existence T_c and T_u, there exist structural matrices $\mathbf{B}^c$ and $\mathbf{B}^u$. There are also Parikh's vectors $\mathbf{v}_k^c$ and $\mathbf{v}_k^u$. When $t_i, t_j, t_k, \ldots$ is a firing transition sequence, corresponding Parikh's vector $\mathbf{v}_k = \mathbf{v}_k^c + \mathbf{v}_k^u$ where a $\mathbf{v}_k^c$ component $v_k^c = v(i)$ if t_i is measurable, otherwise $v_k^c = 0$, and a $\mathbf{v}_k^u$ component $v_k^u = v(i)$ if t_i is unmeasurable, otherwise $v_k^u = 0$. Because of the existence of the P_m and P_{nm} the output vector $\mathbf{y}_k$ has to be introduced. Namely, only measurable states are observable from outside. Thus,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{B}^c.\mathbf{u}_k^c + \mathbf{B}^u.\mathbf{u}_k^u; \quad k = 0, 1, \ldots, N, \tag{6}$$

$$\mathbf{y}_k = \mathbf{C}.\mathbf{x}_k. \tag{7}$$

To deal with such a problem, there are several approaches how to do it. Some other kinds of PN exist which are more suitable for DES control. Moreover, they utilize the P/T PN model presented above. In order to present how they can be used in this direction, let us introduce some of them: Controlled PN, Labelled PN and Interpreted PN.

For the completeness' sake, it is necessary to bring to mind the above mentioned TPN suitable to analyse performance evaluation – see the author's papers [7, 8] where TPN were applied. Moreover, we must not forget Hybrid PN (HPN), more precisely First Order HPN (FOHPN), suitable for modelling hybrid systems containing a continuous part and a discrete-event part, that were also applied in [7, 8].

For these reasons given, these two kinds of PN are not mentioned in details here, in this paper.

2.1 Controlled Petri Nets

The P/T PN described above can be influenced from outside by means of external interferences and/or controller in order to affect the movement of tokens. In such a case we are speaking about Controlled Petri Nets (CtPN) – see e.g. [36, 32, 28, 29, 30, 31]. While P/T PN can be formally represented by the triple $N = \langle P, T, B\rangle$, CtPN can be expressed as the triple

$$PNc = \langle N, P_c, B_c\rangle \tag{8}$$

with P_c being a finite set of control places where $P_c \cap P = \varnothing$, $P_c \cap T = \varnothing$, and $B_c \subseteq P_c \times T$ being the set of directed arcs from control places to P/T PN transitions. In CtPN the places $p \in P$ are named as state places. The set of control places entering a transition $t \in T$ can be expressed as ${}^{(p_c)}t = \{p_c|(p_c, t) \in B_c\}$. On the other hand for a control place $p_c \in P_c$ the set of its output transitions can be denoted as $p_c^{(t)} = \{t|(p_c, t) \in B_c\}$. Denote the set of all controlled transitions by T_c.

A control for a CtPN is understood to be a function $u : P_c \to \{0, 1\}$. It assigns a binary value to each control place. All such controls are denoted by $\mathcal{U}$. They influence a set of transitions $\mathcal{T} \in T$. When for all $t \in \mathcal{T}, u(p_c) = 1$ for all $p_c \in {}^{(p_c)}t$ the set $\mathcal{T}$ is said to be control enabled. An example of CtPN is displayed in Figure 4. In the vector/matrix expression, CtPN can be described as follows:

$$\begin{pmatrix} \mathbf{x}_{k+1} \\ \mathbf{x}^c_{k+1} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_k \\ \mathbf{x}^c_k \end{pmatrix} + \begin{pmatrix} \mathbf{B} \\ \mathbf{B}_c \end{pmatrix}.\mathbf{u}_k, \quad k = 0, 1, \ldots, N, \tag{9}$$

$$\begin{pmatrix} \mathbf{F} \\ \mathbf{F}_c \end{pmatrix}.\mathbf{u}_k \leq \begin{pmatrix} \mathbf{x}_k \\ \mathbf{x}^c_k \end{pmatrix}, \tag{10}$$

$$\mathbf{x}^{ct}_{k+1} = \mathbf{x}^{ct}_k + \mathbf{B}_{ct}.\mathbf{u}_k, \quad k = 0, 1, \ldots, N, \tag{11}$$

$$\mathbf{F}_{ct}.\mathbf{u}_k \leq \mathbf{x}^{ct}_k \tag{12}$$

where $\mathbf{x}^{ct}_k = (\mathbf{x}^T_k (\mathbf{x}^c_k)^T)^T$, $\mathbf{B}_{ct} = (\mathbf{B}^T \mathbf{B}^T_c)^T$, $\mathbf{F}_{ct} = (\mathbf{F}^T \mathbf{F}^T_c)^T$, $\mathbf{G}^T_{ct} = (\mathbf{G}\mathbf{G}_c)^T$, $\mathbf{B} = \mathbf{G}^T - \mathbf{F}$ and $\mathbf{B}_c = \mathbf{G}^T_c - \mathbf{F}_c$. Because $\mathbf{F}$ and $\mathbf{G}^T$ are the same as in (3), only the incidence matrices $\mathbf{F}_c$ and $\mathbf{G}^T_c$ are introduced as follows:

$$\mathbf{F}_c = \begin{pmatrix} 0&0&0&1&0&1&0&0&0&0 \\ 0&0&0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0&0&0 \end{pmatrix}; \quad \mathbf{G}^T_c = \begin{pmatrix} 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \end{pmatrix}. \tag{13}$$

However, because the control places have no input transitions, $\mathbf{G}^T_c = \mathbf{0}$. It means that the firing of transitions does not contribute to the marking development of the control places.

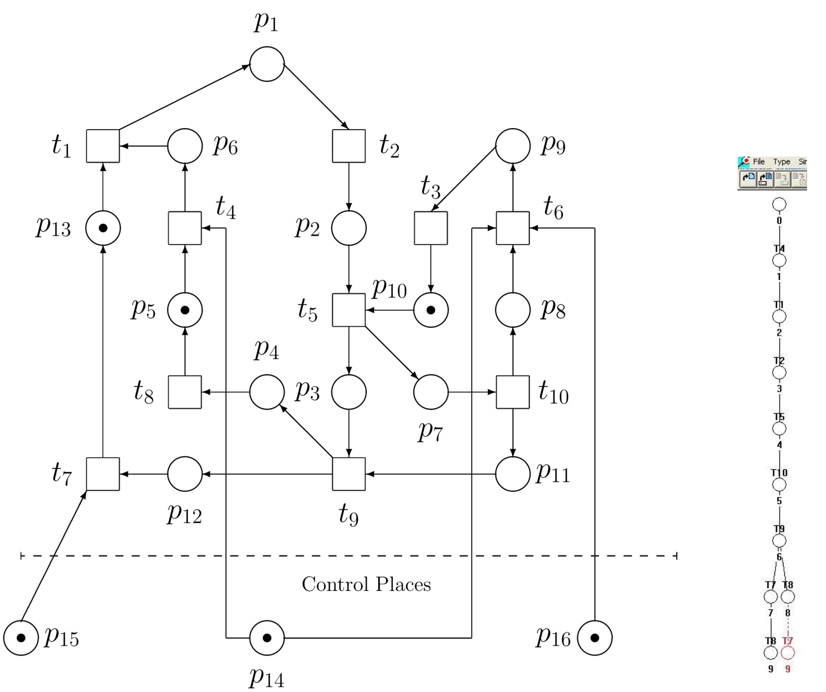

Figure 4. An example of CtPN (left) and its RT (right)

At the initial state $\mathbf{x}_0 = (0\,0\,0\,0\,1\,0\,0\,0\,0\,1\,0\,0\,1\,1\,1\,1)^T$ the particular states of the state space (i.e. the nodes of RT) are the columns of the following matrix:

$$\mathbf{X}_r = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ - & - & - & - & - & - & - & - & - & - \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}. \qquad (14)$$

However, a suitable setting of the control place markings from outside, e.g. by means of a suitable controller, makes possible to control the model of the system. In such case the control places permanently represent an output of the controller.

Over time many approaches how to synthesize controller and/or supervisor for DES control were developed. There exist different kinds of the supervisor synthesis – see e.g. [61, 44, 45, 46, 50, 19, 66, 33, 34] and many others included also in [9, 7, 27].

2.1.1 An Example

Consider e.g. the group of five simple autonomous agents $GrA = \{A1, A2, A3, A4, A5\}$ with the same structure expressed by a working cycle $\{p_i, t_i, p_{i+1}, t_{i+1}\}$ – see Figure 5 – where particular places and transitions mean: p_i – the agent is idle; p_{i+1} – the agent is working; t_i – the agent is starting the work; t_{i+1} – the agent is ending the work.

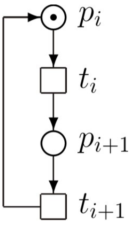

Figure 5. An example of the P/T PN

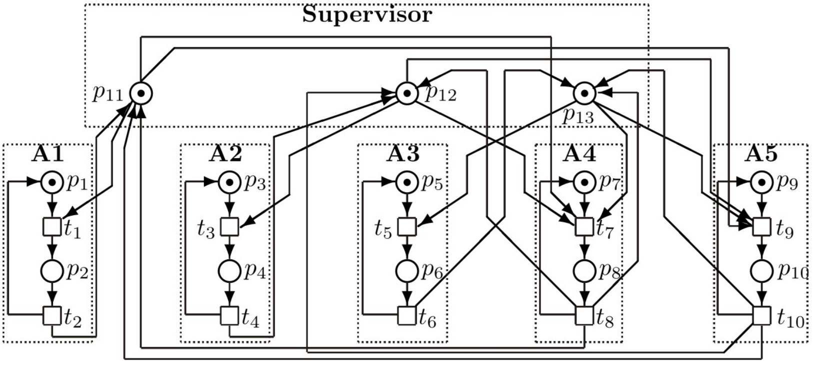

Figure 6. An example of the CtPN

Let us solve the situation when it is necessary to ensure that only one agent from the subgroup $Sgr1 = \{A1, A4, A5\}$ (a representative), only one agent from the subgroup $Sgr2 = \{A2, A4, A5\}$, and only one agent from the subgroup $Sgr3 = \{A3, A4, A5\}$ can simultaneously cooperate with other agents from GrA. In other words, the agents inside the designated subgroups must not work simultaneously.

Even, the agents $A4$ and $A5$ can work only individually (any cooperation with other agents is excluded). However, the agents $A1$, $A2$, $A3$ can work simultaneously. The conditions prescribing the cooperation of agents are:

$$\sigma_{p_2} + \sigma_{p_8} + \sigma_{p_{10}} \leq 1, \tag{15}$$

$$\sigma_{p_4} + \sigma_{p_8} + \sigma_{p_{10}} \leq 1, \tag{16}$$

$$\sigma_{p_6} + \sigma_{p_8} + \sigma_{p_{10}} \leq 1. \tag{17}$$

After applying the method of the supervisor synthesis (based on P-invariants of PN) presented in [7, 9] the PN model of the cooperating agents is displayed in Figure 6. As it can be seen, the supervisor (controller in DES relations) created by p_{11}, p_{12}, p_{13} coordinates the activities of the five agents. In other words, it becomes the additional sixth agent enabling the prescribed cooperation of the five agents.

2.1.2 Petri Nets Controllability and Observability

In PN the terms controllability and observability are nearly related. As to observability, the so called silent transitions [10, 13] are invisible in an environment and may cause problems with observability of discrete events. They represent the discrete events that cause a change in the state of the DES, however they are not observable from outside – i.e., they become unobservable. Therefore, it is particularly important to obtain a powerful approach to PN-based DES control which relies only on information about the observable transitions and forbids firing the unobservable ones. When a controller is not allowed to influence some transitions, such transitions become uncontrollable. Both such groups of transitions bring uncertainty into the DES control. Therefore, it is necessary to deal with the problem how to control DES in such uncertainty conditions.

More precisely said, a transition is named to be *unobservable* when its firing cannot be directly detected or measured. A transition is named to be *uncontrollable* when its firing cannot be inhibited by an external action. All unobservable transitions are implicitly uncontrollable [43, 44].

In [24, 47] controllability in the framework of Petri nets is mentioned. Unfortunately, the necessary and sufficient conditions being able to decide about controllability and observability are known only in continuous and/or discrete-time linear systems. In continuous linear time invariant systems on the controllability of a system with the state equation $\dot{\mathbf{x}} = \mathbf{A}.\mathbf{x} + \mathbf{B}.\mathbf{u}$, $\mathbf{x} \in \mathbb{R}^n$ (where $\mathbb{R}$ represents real numbers), $\mathbf{u} \in \mathbb{R}^m$ decides – see e.g. [38, 20, 48] – the *rank* of the controllability matrix $\mathbf{CM} = [\mathbf{B}|\mathbf{AB}|\mathbf{A}^2\mathbf{B}|\cdots|\mathbf{A}^{n-1}\mathbf{B}]$ with the dimensionality $(n \times n.m)$. The system is controllable when $rank(\mathbf{CM}) = n$. This is valid not only for continuous linear systems but also for discrete-time modification of linear time invariant systems. A system is called controllable (or reachable) if all states are reachable (i.e. the reachable set $R \in \mathbb{R}^n$). The uncontrollable system in which uncontrollable part is stable is named the stabilizable system.

When the output equation of the continuous linear time invariant systems is $\mathbf{y} = \mathbf{C}.\mathbf{x} + \mathbf{D}.\mathbf{u}$, $\mathbf{y} \in \mathbb{R}^p$, the observability is decided on the *rank* of the observability matrix $\mathbf{OM} = [\mathbf{C}^T|\mathbf{A}^T\mathbf{C}^T|(\mathbf{A}^T)^2\mathbf{C}^T|\cdots|(\mathbf{A}^T)^{n-1}\mathbf{C}^T]$. The system is observable when $rank(\mathbf{OM}) = n$. In other words, a system is observable if the initial state can be obtained (observed) from the knowledge of the input and the output. Unobservable system in which the unobservable subsystem is stable is named as the detectable system.

However, in PN modelling the completely different kind of systems, i.e. DES, the *rank* of such controllability matrix $\mathbf{M}_c = (\mathbf{B}|\mathbf{B}|\cdots|\mathbf{B})$ of the system (1) always coincides with the *rank* of the PN incidence matrix $\mathbf{B}$, because in (1) $\mathbf{A} = \mathbf{I}$ (i.e. the identity matrix). Therefore, $rank(\mathbf{B}) = n$ is only a necessary condition for controllability if the control input is restricted to $\mathbf{u}_k \in \{0,1\}^m$ and to $\mathbf{x}_k + \mathbf{B}.\mathbf{u}_k \geq \mathbf{0}$. Moreover, the PN state equation (1) itself only provides a necessary but not sufficient condition for reachability.

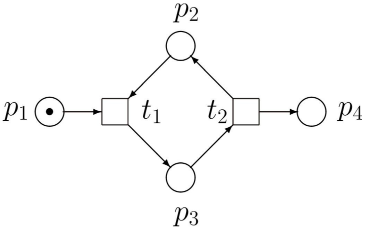

Figure 7. An example on the controllability of P/T PN

For example, consider the net system in Figure 7 with the structure

$$\mathbf{F} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}; \quad \mathbf{G}^T = \begin{pmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}; \quad \mathbf{B} = \mathbf{G}^T - \mathbf{F} = \begin{pmatrix} -1 & 0 \\ -1 & 1 \\ 1 & -1 \\ 0 & 1 \end{pmatrix} \qquad (18)$$

and initial state $\mathbf{x}_0 = (1,0,0,0)^T$. Although the $rank(\mathbf{B}) = 2$, even for $\mathbf{u}_0 = (1,1)^T$ when $\mathbf{x}_0 + \mathbf{B}.\mathbf{u}_0 \geq \mathbf{0}$ the state $\mathbf{x}_1 = (0,0,0,1)^T$ is not reachable. In fact, no transition is enabled at the initial state. Namely, neither the sequence t_1t_2 nor the sequence t_2t_1 can be fired because they are not enabled. It means practically that no RT exists (more precisely, RT consists of only one node $\mathbf{x}_0$ and has no arcs).

Still more complicated situation than that in case of the controllability is the situation with the observability of PN. Because $\mathbf{A} = \mathbf{I}$ the *rank* of the observability matrix $\mathbf{M}_o = (\mathbf{C}^T|\mathbf{C}^T|\cdots|\mathbf{C}^T)$ is reduced on the *rank* of the matrix $\mathbf{C}^T$, which principally cannot have the $rank(\mathbf{C}^T) = n$. Really, the number of measurable places $p < n$, where n is the total number of places.

Only very few published works appertain to observability in PN – see e.g. [52, 25, 22, 54, 65]. In [65] CtPN model is used for forbidden state avoidance under partial event observation with the assumption that the initial marking is known. The observability properties depend not only on the net structure N, but also on the initial marking m_0, that in [22] is assumed to be unknown. The structural observability and marking observability are distinguished.

The structural observability requires the study of the system properties for all possible initial markings. If a place $p \in P$ is observable in $\langle N, M \rangle$ then it is also observable in $\langle N, M' \rangle$ $\forall M' \geq M$ with $M'(p) = M(p)$.

The marking observability means that there exists at least one word that is complete, while strong marking observability means that all words can be completed in a finite number of steps into a complete word. Here, the term *word* means a word of events – i.e. a sequence of transition firings. It has the direct relationship with RG and/or CG. Observability in general [54] allows, through an observer, the computation of state variable values that cannot be directly measured. Observers are used to estimate the system state.

Finally, it has to be said that for control of PN with unobservable/uncontrollable transitions and non-measurable places CtPN are not too suitable. Consequently, it is necessary to apply next kinds of PN mentioned below, i.e. Labelled PN and Interpreted PN.

2.2 Labelled Petri Nets

Labelled Petri nets (LbPN) are the standard Petri nets with a function attaching a label to each event. As a fundamental operation in modular design the parallel composition LbPN is used. Often, models of subsystems are combined into a model of the whole system. PN languages (at least their simple forms) are applied in LbPN.

In terms of a separate set of event labels, PN languages were defined e.g. in [14, 35, 37, 40, 47, 49, 63], etc. Events can be assigned to some transitions, even to all of the transitions. Then, the firing of a transition represents an event labelled by the corresponding label. The set of all sequences of admissible event is named as PN language.

In general, the formal definition of LbPN is the following:

$$LbPN = \langle N, \mathcal{L}, l, m_0, F_m \rangle . \tag{19}$$

Here N is the PN structure; $\mathcal{L} = L \cup \varepsilon$ is an alphabet representing a finite set of events, where L represents observable events and ε represents unobservable events; $l : T \rightarrow \mathcal{L}$ is a labeling function assigning an event to each transition; $m_0 \in \mathcal{M}$ is an initial marking, with $\mathcal{M}$ being a set of markings; $F_m \in \mathcal{M}$ is a finite set of final markings.

At the simplest understanding of LbPN – in the sense of the λ-free PN language [49] – no transition is labelled with the empty string λ and two or more transitions may have the same label. Otherwise, especially in case of so called silent events [23, 4] being unobservable and causing a change of DES markings (states), transitions may be labelled with the empty string ε. In case of indistinguishable events, that may yield two or more new states from a given state, two or more transitions are labeled with the same symbol and enabled at a given state.

2.2.1 State Estimation

In linear time invariant systems (continuous and/or discrete-time) in case of presence of the state disturbance or noise and sensor noise or error which are described statistically, or assumed to be small, when $\mathbf{u}(t)$ and $\mathbf{y}(t)$ are observed on an interval $(0, t-1)$ the problem of the state estimation is the following. To estimate $\mathbf{x}(s)$ from $\mathbf{u}(t)$ and $\mathbf{y}(t)$ three possibilities exist:

1. to estimate the initial state ($s = 0$);
2. the current state ($s = t - 1$);
3. to estimate (i.e. predict) next state ($s = t$).

The algorithm or a system yielding the estimate $\hat{\mathbf{x}}(s)$ is called an observer or state estimator.

In PN the situation is different, likewise as in case of the controllability and observability. The state estimation depends on the so called PN sensors.

2.2.2 Sensors in PN

The model of PN with outputs consists not only from the state equation but also from the output equation $\mathbf{y}_k = \mathbf{C}.\mathbf{x}_k$. PN with outputs [2, 56] are PN with the so called *place sensors* which count the tokens contained in some places – named as measured or observable places (in modelled DES e.g. vision sensors, touch sensors, etc.), and *transition sensors* which detect the firing of some of the transitions – named as measured or observable transitions (in real DES e.g. motion sensors, speed sensors, etc.). However, not all places and not all transitions may have a sensor. As it was premised above, in modelled DES not all state variables and not all control variables can be measured or observed. More precisely, a place sensor configuration V is a vector $(v_1, v_2, \ldots, v_n)^T$, where $v_i = 0$ if no place sensor exists on place p_i and $v_i = 1$ otherwise. $|V| = \sum_{i=1}^{n} v_i \leq n$ denotes the total number of place sensors in the place sensor configuration V. On the other hand, analogically, $|L| \leq m$ is the total number of transition sensors in use and could be zero if no transition sensor is used.

As it was already mentioned, LbPN may have nondeterministic transitions – i.e. transitions that share the same label or unobservable transitions associated with the null label. Other faults are modelled also as unobservable transitions.

The set of transitions $T = T_d \cup T_n$, where T_d is a set of deterministic transitions, while T_n is the set of nondeterministic transitions. For deterministic PN with the so called λ-free labeling function it is defined [23] that if two transitions are labelled with the same symbol they cannot simultaneously be enabled at $\mathcal{M}$. Therefore, for structurally deterministic function two different transitions cannot be labelled with the same symbol. Thus, a transition t is nondeterministic if its label is associated also to other transitions, otherwise a transition t is said to be deterministic. The association between sensors and transitions can be captured by a labeling function $\mathcal{L} : T \to L \cup \{\varepsilon\}$ that assigns to each transition $t \in T$ either a symbol from a given alphabet L or the empty string ε. The set of transitions whose label is ε is denoted as $T_u = \{t \in T | \mathcal{L}(t) = \varepsilon\}$. Transitions $t \in T_u$ are called unobservable or silent. T_o denotes the set of transitions labelled with a symbol in L. Transitions $t \in T_o$ are called observable, because when they fire, their label can be observed. In general, the same label $l \in L$ can be associated to more than one transition. In particular, two transitions $t_1, t_2 \in T_o$ are called non-distinguishable if they share the same label, i.e., $L(t_1) = L(t_2)$. The set of transitions sharing the same label l are denoted as T_l.

In LbPN whose initial marking is not known exactly [5], marking estimation is possible. It is sufficient to know only that the marking belongs to a given convex set. The silent transitions (i.e. labelled with the empty word) and indistinguishable transitions (i.e. sharing the same label with other transitions) are allowed on that way.

For the partially observed LbPN with a place sensor configuration V, a labeled function $\mathcal{L}$ and an *observation sequence*, the set of *consistent firing sequences* and *consistent markings* can be found [55]. The firing of t will generate token changes in place sensors and/or an observable transition label.

2.2.3 Illustrative Example 1

Consider a partially observed P/T PN given in Figure 8 (left), where t_4, t_5 are unobservable.

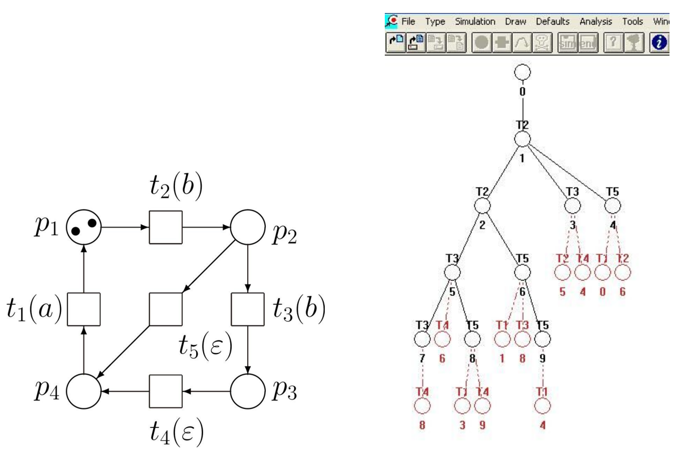

Figure 8. An example of partially observed LbPN (left) and the RT corresponding to the fully observable net (right)

Consequently, p_4 is unobservable too because both of its input transitions are unobservable. Thus, $P_o = \{p_1, p_2, p_3\}$ is the set of observable places while $P_{uo} = \{p_4\}$ is the set (singleton) of unobservable ones. Consider that only p_2 is equipped by a sensor. Next, $T_o = \{t_1, t_2, t_3\}$ is the set of observable transitions while $T_u = \{t_4, t_5\}$ is the set unobservable ones. The labeling function $\mathcal{L}$ is given as $\mathcal{L}_1 = a$, $\mathcal{L}_2 = \mathcal{L}_3 = b$, $\mathcal{L}_4 = \mathcal{L}_5 = \varepsilon$.

An observable transition $t \in T_o$ can have a sensor that indicates when a transition within a given subset of transitions has fired. Any unobservable transition $t \in T_u = T - T_o$ cannot have such a sensor associated with it. The association between sensors and transitions is captured by a labeling function $\mathcal{L} : T \to L \cup \{\varepsilon\}$ that assigns a label to each transition. Unlike the fixed labeling function in LbPN, the labeling function $\mathcal{L}$ in a partially observed PN can be configured subject to the constraint that unobservable transitions must be assigned the null label – $\mathcal{L}(t) = \varepsilon$ for all $t \in T_u$.

In case of fully observable net, its RT has the form as it is displayed in Figure 8 (right) with the nodes represented by the columns of the matrix

$$\mathbf{X}_r = \begin{pmatrix} 2 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 2 \end{pmatrix}. \tag{20}$$

LbPN are suitable for testing the observability of places and transitions. For example let the transition t_5 be the only fault transition which needs to be detected. Firing the sequence of transitions $S = t_2 t_5$ the system trajectory is $M0[t2 > M1[t5 > M4$, where $M0 = (2000)^T$, $M1 = (1100)^T$ and $M4 = (1001)^T$, and the corresponding observation from place and transition sensors is $M0 \to b \to M1 \to \varepsilon \to M4 \to a \to M0$. Hence, we can deduce that the fault transition t_5 had to be occurred. Namely, only the firing of t_5 can decrease the number of tokens in p_2 by 1 and at the same time not generate any label – b in t_3. The observation is driven by token changes in observable places with sensors and/or observed labels. When the observed label is ε, it means that there is no observation output from transition sensors. Because of the unobservable transitions and the unobservable place, the actual marking of p_4 and actual firing of t_4, t_5 cannot be anticipated (they are unpredictable).

2.2.4 Illustrative Example 2

To illustrate LbPN deeper, consider the example of such a net given in Figure 9.

There are 13 transitions in the net. Consider that the observable transitions create the set $T_o = \{t_1, t_2, t_6, t_7, t_8, t_9, t_{12}\}$ while the unobservable ones create the set $T_u = \{t_3, t_4, t_5, t_{10}, t_{11}, t_{13}\}$, i.e. $T_u = \{\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4, \varepsilon_5, \varepsilon_6\}$. Let the labeling functions of the observable transitions be $\mathcal{L}(t_1) = a$, $\mathcal{L}(t_2) = \mathcal{L}(t_6) = b$, $\mathcal{L}(t_7) = \mathcal{L}(t_8) = c$, $\mathcal{L}(t_9) = \mathcal{L}(t_{12}) = d$. RT of the fully observable net is given in Figure 10.

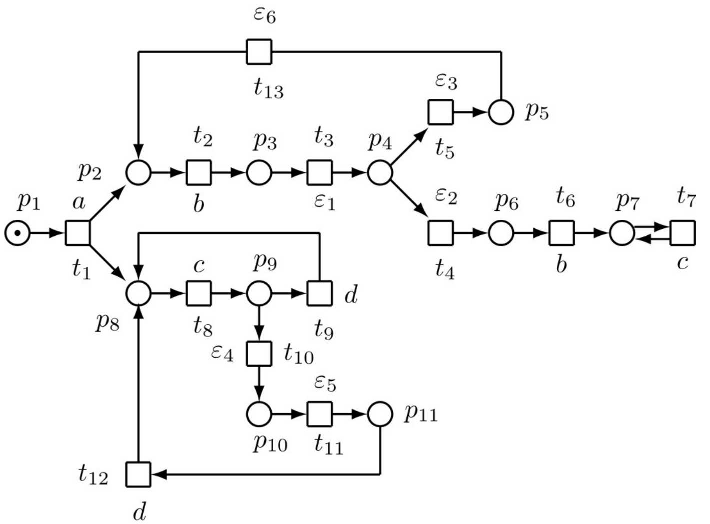

Figure 9. An example of LbPN

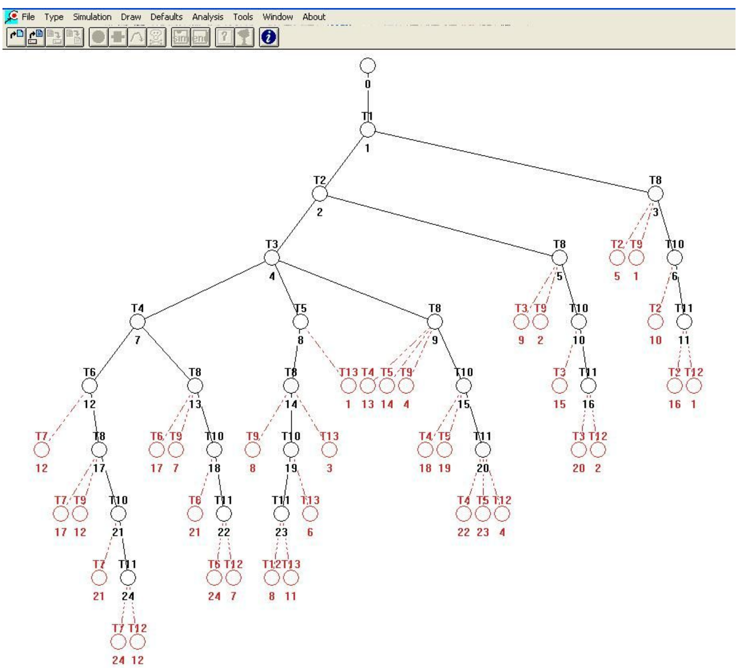

Figure 10. The RT of the corresponding P/T PN

It has 25 nodes being columns of the matrix where the first column represents the initial marking M_0.

$$\mathbf{X}_r = \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&1&0&0&1&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&0&0&1&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&1&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&1&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&1&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&1&0&0&0&1&0&0&1 \\ 0&1&1&0&1&0&0&1&1&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&1&0&1&0&0&0&1&0&0&0&1&1&0&0&1&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0&0&0&1&0&0&0&0&1&0&0&1&1&0&1&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&1&0&0&0&1&0&1&1&1 \end{pmatrix}. \tag{21}$$

When we consider the input word $w = ab$, we obtain the set of consistent firing sequences in the form $S(w) = \{t_1t_2, t_1t_2\varepsilon_1, t_1t_2\varepsilon_1\varepsilon_2, t_1t_2\varepsilon_1\varepsilon_2\varepsilon_6, t_1t_2\varepsilon_1\varepsilon_3\}$. In RT they cause the set of consistent markings M_1, M_2, M_4, M_8 and M_7 represented by the columns 2, 3, 5, 9, 8 of the reachability matrix $\mathbf{X}_r$, respectively. When the word is considered to be $w = acd$, then $S(w) = \{t_1t_8t_9, t_1t_8\varepsilon_4\varepsilon_5t_{12}\}$, the set of markings is represented by the column 2. Both cases of the word w are displayed in Figure 11 – the former word on the left branch of the cropped RT, while the latter one on the right branch.

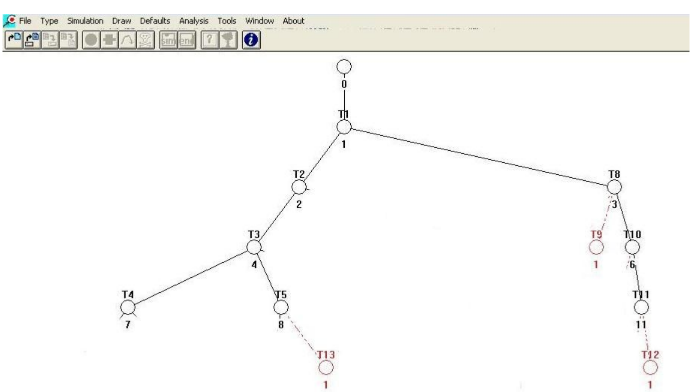

Figure 11. The cropped RT expressing the reachable markings corresponding to the both words – on the left side to the word $w = ab$ and on the right side to the word $w = acd$

2.3 Interpreted Petri Nets

IPN can be formally described [1, 39, 54, 57, 58, 8] by the quadruplet as follows:

$$IPN = \langle N, \Sigma, \Phi, \lambda, \Psi, \varphi \rangle \tag{22}$$

where N is the PN structure; $\Sigma = \{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ is an input alphabet with α_r, $i = 1, 2, \ldots, r$ being input symbols; $\Phi = \{\delta_1, \delta_2, \ldots, \delta_s\}$ is the output alphabet with δ_i, $i = 1, \ldots, s$, being the output symbols; $\lambda = T \rightarrow \Sigma \cup \{\varepsilon\}$ is the labeling function of transitions (assigning either an input symbol $\alpha_i \in \Sigma$ or the internal event ε to each transition) with the constraint: $\forall t_j, t_k \in T$, $j \neq k$, if $\forall p_i\, F(p_i, t_j) = F(p_i, t_k) \neq 0$ and both $\lambda(t_j) \neq \varepsilon$, $\lambda(t_k) \neq \varepsilon$ then $\lambda(t_j) \neq \lambda(t_k)$ – i.e. each transition is assigned a unique label; $\Psi : P \rightarrow \Phi \cup \{\varepsilon\}$ labels the places (either an output symbol $\delta_i \in \Phi$ or the null output signal ε is assigned to each PN place by this function); $\varphi : R(IPN, M_0) \rightarrow \mathbb{Z}_{\geq 0}^{q \times n}$ is an output function that associates an output vector to each marking $R(IPN, M_0)$. Here $q \in \mathbb{Z}_{>0}$ is a positive integer representing the number of available output signals and $n = |P|$ – i.e. the number of all places in the set P. It means that φ is $(q \times n)$ matrix.

IPN are to model the DES behavior that includes partially observable both events and states [53]. The net system where each transition is assigned a unique label is named as free-labeled PN. Identifying of such nets is possible. The approaches consist in observing the marking of a subset of places and (when some additional information on the dependency between transitions is given) allow to reconstruct the part of the net structure related to unobservable places. Such approaches can be found e.g. in [41, 42] and the approach based on integer programming in [26].

3 IPN IN MODELLING AND CONTROL OF DES

The control specifications create [21] a set of forbidden markings M_F which correspond to undesirable states. Namely, they either jeopardize the system safety or they give birth to deadlock situations. Therefore, it is necessary to determine a convenient set of places that, after adding to the PN model of the plant, will prevent the whole system from reaching these states. When the PN model has uncontrollable transitions it is necessary to prevent the system from reaching the forbidden markings, containing all dangerous markings from which a forbidden one may be reached by firing a sequence of uncontrollable transitions from T_u.

Let M_D be the set of dangerous markings, i.e. $M_D = \{M \in R(N, M_0) \mid \exists M' \in M_F \wedge \tau \in T_u^*, M[\tau > M'\}$, where T_u^* is a sequence of uncontrollable transitions. Of course, $M_F \subseteq M_D$. The controlled system has to be safe (without forbidden markings) and live (without deadlocks). Then, the set M_L of legal (i.e. admissible) markings is the maximal set of reachable markings such that

1. $M_L \cap\ M_D = \varnothing$;

2. it is possible to reach the marked marking M_0 from any legal marking without leaving the set M_L;
3. any transition t from a legal marking to a nonlegal marking is a controllable transition.

Of course the full RG is cropped into R_c being the RG containing all legal markings. Thus, $M_L \subseteq R(N, M_0) \setminus M_D$.

It was shown in [50] that M_L exists and it is called the *maximally permissive behavior*. M_L is such that, whatever the marking in M_L, the system cannot be uncontrollably led outside M_L.

IPN are closely related to LbPN. Even, it may be said that IPN are a modification of LbPN. They are [51] an extension of PN that allow to represent *the output signals generated when a marking is reached*, and *the input signals associated with transitions that are controllable*. These signals are useful to infer the initial marking of the net. Such a kind of PN is also suitable to model and control DES the PN model of which contains uncontrollable transitions and unmeasurable places.

Let us introduce two illustrative examples concerning

1. the principle of creating the IPN model of DES at the existence of uncontrollable transitions and unmeasurable places;
2. the principle of its control.

3.1 Example – Creation of IPN Model

To show how to create the IPN model let us consider the simple PN-model given in Figure 12.

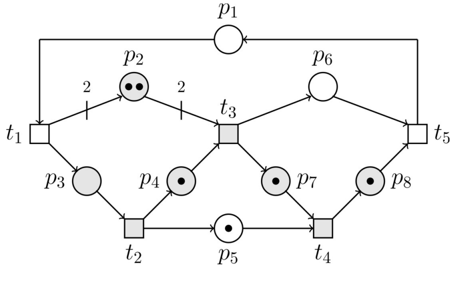

Figure 12. An example of the PN-based model

The full RT (of the net without taking unmeasurable places and uncontrollable transitions into account) is displayed in Figure 13 left. The particular RT nodes create the columns of the following matrix:

$$\mathbf{X}_r = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 2 & 0 & 0 & 0 & 2 & 2 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 2 & 2 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 2 & 0 & 1 & 2 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 2 & 2 & 0 & 1 & 0 & 1 & 0 & 0 \end{pmatrix}. \tag{23}$$

Suppose that the measured places are $P_m = \{p_1, p_5, p_6\}$ and the unmeasured ones (filled by the gray color) are $P_{um} = P \backslash P_m = \{p_2, p_3, p_4, p_7, p_8\}$. Consider that the controllable transitions are $T_c = \{t_1, t_5\}$ while the uncontrollable ones (filled by the gray color) are $T_u = T \backslash T_c = \{t_2, t_3, t_4\}$. Consider for such IPN the input alphabet $\Sigma = \{a, b\}$ and the output alphabet $\Phi = \{\delta_1, \delta_2, \delta_3\}$. Consequently, $\lambda(t_k)_{k=1,\ldots,5} = \{a, \varepsilon, \varepsilon, \varepsilon, b\}$, $\Psi(p_i)_{i=1,\ldots,8} = \{\delta_1, \varepsilon, \varepsilon, \varepsilon, \delta_2, \delta_3, \varepsilon, \varepsilon\}$. The output equation is as follows:

$$\mathbf{y}_k = \varphi.\mathbf{x}_k \qquad \text{where} \qquad \varphi = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}. \tag{24}$$

It is necessary to say that the RT of such IPN is not connected in spite of the fact that t_1, t_5 are controllable, because we do not know what is up with the marking development inside of the *foggy area* (containing uncontrollable transitions and states containing unmeasurable places – see the right side of Figure 13).

Because at the beginning p_1 is passive ($\sigma_{p_1} = 0$) and p_5 is active ($\sigma_{p_5} = 1$), the initial state is $\mathbf{x}_0 = (0, \varepsilon, \varepsilon, \varepsilon, 1, 0, \varepsilon, \varepsilon)^T$. Neither t_1 nor t_5 are enabled (regardless of the activity or passivity of unmeasurable places p_2 and p_8, respectively) because measurable places p_1, p_6 are passive (without tokens).

On the other hand for the initial state $\mathbf{x}_0 = (0, 2, 0, 1, 1, 0, 1, 1)^T$ the output of the PN (without taking the influence of uncontrollable transitions into account) is $\mathbf{y}_0 = (0, 1, 0)^T$.

3.2 Example – Principle of IPN Model Control

To explain how the IPN model is controlled, consider a segment shown in Figure 14 left. While the upper *line* containing p_4 and t_3 represents the fragment of the model of the control system PN_{cs} (containing the control specifications), the lower line represents the fragment of the IPN model of the controlled plant PN_{pl}. The RG of the model is given in Figure 14 right, where $\mathbf{x}_0 = (1, 0, 0, 1)^T$, $\mathbf{x}_1 = (0, 1, 0, 1)^T$, $\mathbf{x}_2 = (0, 0, 1, 1)^T$ and $\mathbf{x}_3 = (0, 0, 1, 0)^T$.

Here, the controllable transition t_1 is enabled because it is required to reach the stated output while p_4 represents the state of a sensor. The self-loop between them represents the relation between the place of the control specification and the

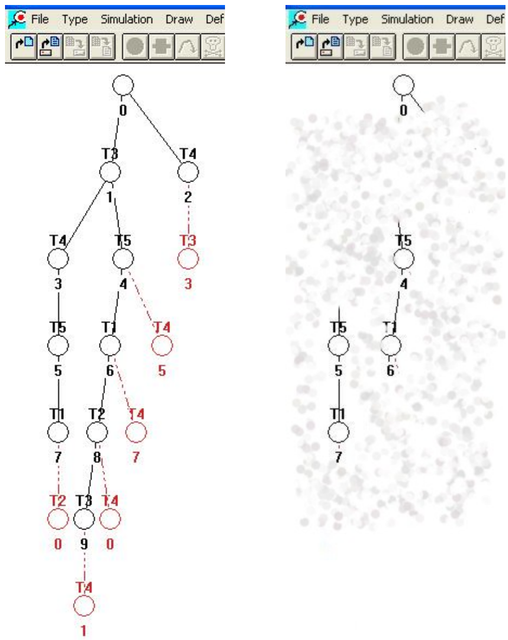

Figure 13. The RT corresponding to the PN-based model (left) and to the IPN model (right)

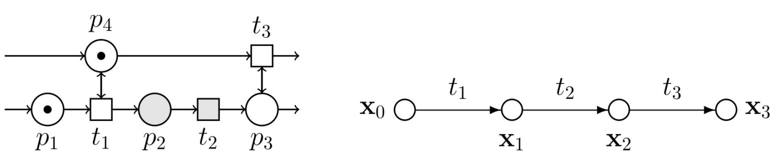

Figure 14. The controlled segment of the IPN-based model (left) and its RG (right)

plant controllable discrete event. The transition t_3 represents enabling the event expressing the situation when the plant and control specification have the same output and p_3 represents the state of the sensor. The self-loop between them expresses the relation between the plant measured place and the control specification. Thus, t_2 is bypassed and it can fire (as an internal event) or not. Note that the fragment of RG accordant with the segment of the controlled plant is straight-lined. Such fragments occur also in more complicated structures. Of course, RG of more complicated structures of the plant models will not be straight-lined.

3.3 Illustrative Example on Robotized Cell

Consider the robotized cell schematically drawn in Figure 15.

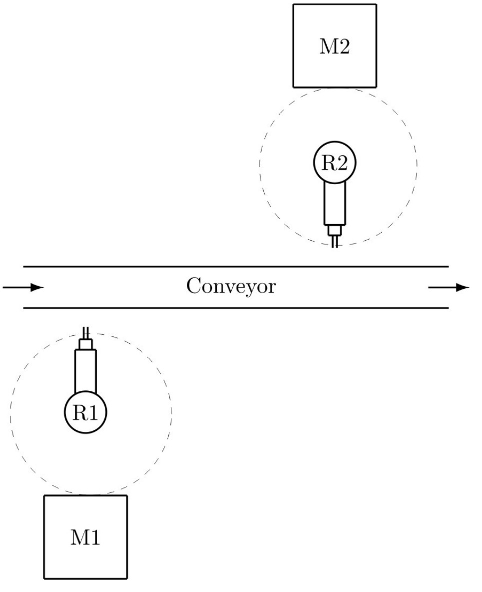

Figure 15. The schema of the plant

There are two robots R1, R2 serving, respectively, machines M1, M2. R1 inserts into M1 a raw material feeded by the Conveyer from an input buffer. M1 machines the raw material. After finishing operations the intermediate product is unloaded from M1 by R1 and put on the Conveyor. By means of the Conveyor the semi-product proceeds towards M2. R2 takes it up and inserts it into M2. After finishing operations the final product is unloaded from M2 by R2 and put on the Conveyor which conveys it to an output buffer. The PN model of the plant operation is given in Figure 16 left. Its RT is given in Figure 16 right. It is very simple, however only in case when all transitions are controllable and all places (states) are measurable. The nodes of the RT are given by rows of the following reachability matrix representing the state vectors $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_8$:

$$\mathbf{X}_r^T = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{pmatrix}. \qquad (25)$$

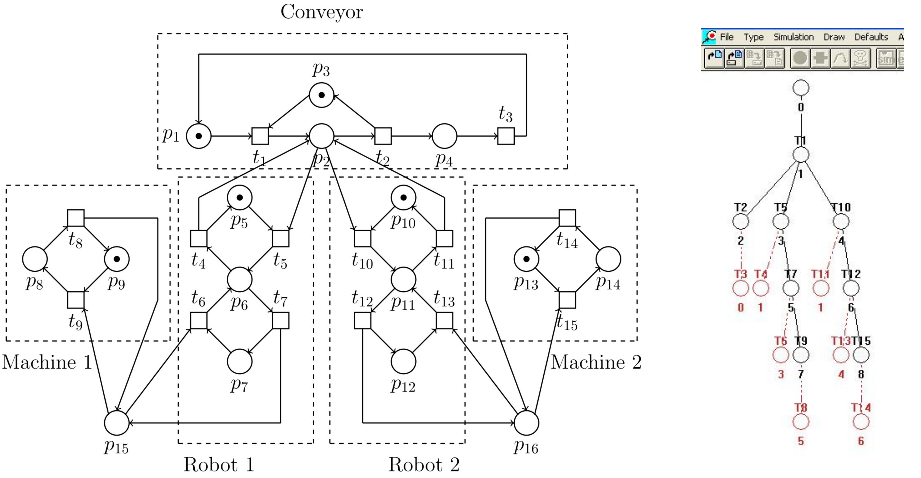

Figure 16. The PN model of the plant (left) and its RT (right)

In opposite case, when there are the uncontrollable transitions and unmeasurable places in the model, the situation is dramatically changed. The IPN model of the plant – see Figure 17 left – has the same structure, but transitions and places filled in gray color complicate the situation. They represent uncontrollable transitions $T_u = \{t_3, t_7, t_8, t_{12}, t_{14}\}$ and unmeasurable places $P_{um} = \{p_4, p_6, p_8, p_{11}, p_{14}\}$. Of course, controllable transitions create the subset $T_c = T \setminus T_u$ and measurable places create the subset $P_m = P \setminus P_{um}$.

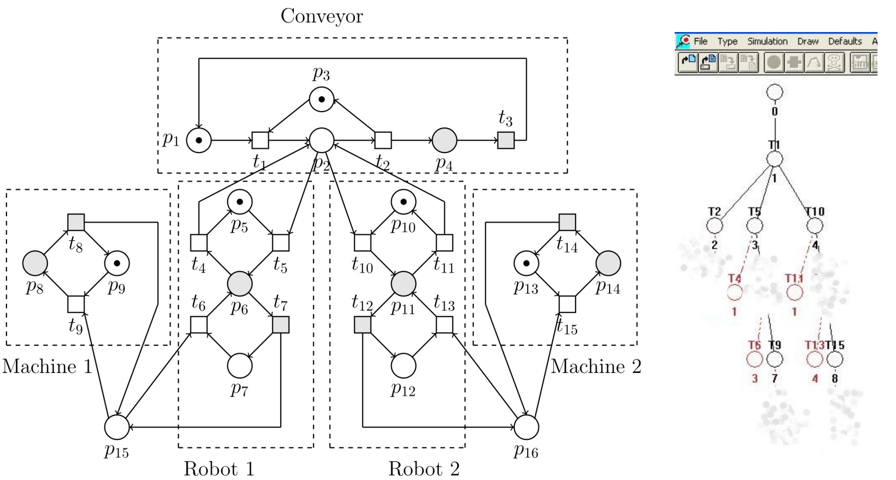

Figure 17. The IPN model of the plant (left) and its RT (left)

In such a case also the previous RT is not clear because of the uncontrollable transitions $t \in T_u$ – see Figure 17 right. Then,

$$\mathbf{X}'_r = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \varepsilon & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & \varepsilon & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \varepsilon & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \varepsilon & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \varepsilon \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}. \tag{26}$$

Because of $p \in P_{um}$ the output vector of such a system is the following:

$$\mathbf{y}_k = \varphi.\mathbf{x}_k \tag{27}$$

where

$$\varphi = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Also in such a case the plant can be controlled by the controller. The structure of the plant with the incorporated controller is in Figure 18.

RT of this model is given in Figure 19. Nodes of the RT are represented by the following reachability matrix

$$\mathbf{X}_r = \begin{pmatrix}
1010100100000000000000000000000000000 \\
0101001000010000000000000000000000000 \\
1010100100000000010000000100000100010 \\
0000000000000000010000000100000100010 \\
1111101100110001110001111001111101111 \\
0000010001000010000010000000000000000 \\
0000000010001100001100000110000010000 \\
0000000000001000001000000010000010000 \\
1111111111110111110111111011111101111 \\
1111111111011110011110001110001100010 \\
0000000000100000100000010000001000000 \\
0000000000000000100000110000110001101 \\
1111111111111111111111011111011110110 \\
0000000000000000000000100000010000101 \\
0000000001000010000010000000100000000 \\
0000000000000000100000010000010000100 \\
1100000000000000000001001101010011000 \\
0011010010001000000000000000001100101 \\
0000101001100101001001000000000000010 \\
0000000010001001011010010010100000000
\end{pmatrix}. \tag{28}$$

At meeting the priorities $t_1 \succ t_{16}$, $t_{17} \succ t_8$, $t_9 \succ t_{17}$, $t_5 \succ t_{17}$, $t_2 \succ t_{19}$, $t_3 \succ t_{19}$, $t_{15} \succ t_{18}$, $t_{18} \succ t_{14}$ corresponding to sequence of technological operations, the finite path (trajectory) ensuring the working cycle of the controlled plant is the following: $\mathbf{x}_0 \xrightarrow{t_1} \mathbf{x}_1 \xrightarrow{t_{16}} \mathbf{x}_3 \xrightarrow{t_5} \mathbf{x}_5 \xrightarrow{t_7} \mathbf{x}_8 \xrightarrow{t_9} \mathbf{x}_{12} \xrightarrow{t_{17}} \mathbf{x}_{18} \xrightarrow{t_8} \mathbf{x}_{13} \xrightarrow{t_6} \mathbf{x}_9 \xrightarrow{t_4} \mathbf{x}_6 \xrightarrow{t_{10}} \mathbf{x}_{10} \xrightarrow{t_{12}} \mathbf{x}_{15} \xrightarrow{t_{15}} \mathbf{x}_{21} \xrightarrow{t_{18}} \mathbf{x}_{27} \xrightarrow{t_{14}} \mathbf{x}_{22} \xrightarrow{t_{13}} \mathbf{x}_{16} \xrightarrow{t_{11}} \mathbf{x}_{11} \xrightarrow{t_2} \mathbf{x}_{17} \xrightarrow{t_3} \mathbf{x}_7 \xrightarrow{t_{19}} \mathbf{x}_0$. Although the uncontrollable transitions $t \in T_u = \{t_3, t_7, t_8, t_{12}, t_{14}\}$ occur in the sequence of the transitions, the control is possible.

4 CONCLUSION

Several kinds of PN and their application to modelling and control of DES were introduced and described. Particularities and curiosities of individual kinds were pointed out, especially with respect to their advisability to model and control of DES. For deterministic PN models, where all transitions are controllable and observable and all places are observable (measurable), the P/T PN, TPN and CtPN are suitable. Also HPN (or FOHPN), mentioned for completeness' sake in the Section 2, are useful.

For PN models of DES where unobservable/uncontrollable transitions (where also transitions modelling failures belong) and unobservable (unmeasurable) places

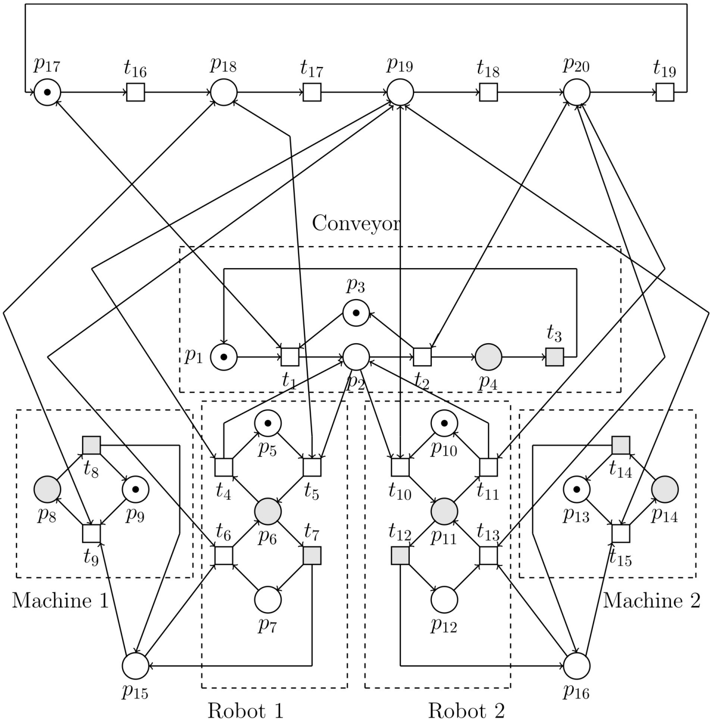

Figure 18. The controlled IPN model of the plant

occur, especially LbPN and IPN are suitable. In both areas of PN models illustrative examples were introduced.

While in the former PN area it is possible to find feasible paths (control trajectories) without big problems, in the latter one this cannot be said. Namely, uncertainty demands completely different approach not only to the model creation in order to express appropriately the uncontrollable/unobservable transitions and unobservable (unmeasurable) places but also to the control synthesis. Though the model is created, it still does not mean that the control synthesis will be simple. Consecutive problems yet occur concerning the controllability and observability of the model which are fundamental over the control synthesis. Moreover, for the state estimation PN sensors are very important as well as the points in PN where they have to be located. Only after the correct dislocation of sensors the successful control can be expected.

For modelling and simulation of DES by means of PN the simulation tools are used. In principle, there are two kinds of simulators:

1. graphical, and
2. computational.

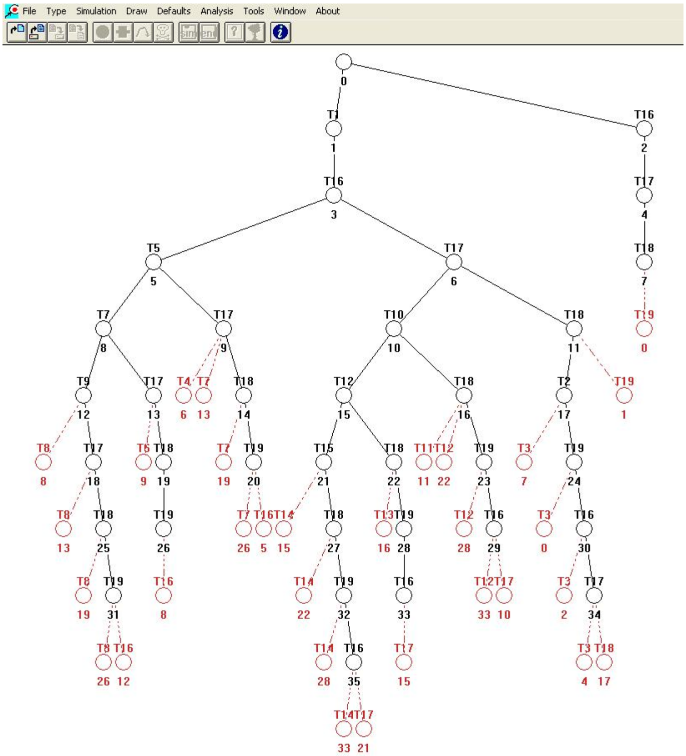

Figure 19. The controlled IPN model of the plant

In case of graphical simulators the PN is drawn by means of clicking on icons (place, transition, arc, token). Then, the marking evolution can be monitored step-by-step by means of clicking on enabled transitions (being distinguished from disabled transitions in color). Some of such simulators are able to draw RT and/or test the basic properties of PN. The graphical simulators are either free or commercial. The list of some available simulators is on the web site `http://www.informatik.uni-hamburg.de/TGI/PetriNets/index.php`. The graphical simulators are applicable for not very large PN models. The computational simulators are built on different bases (C++, Java, etc.). However, for common users the simulators utilizing Matlab tool are most friendly and applicable. Although Matlab contains own PN graphical tool, from computational point of view the simulator HYPENS (suitable for computational simulation of P/T PN, TPN, HPN, FOHPN) [59] used in Matlab is more user friendly. It works with matrix/vector based model for which the Matlab tool is an ideal environment. The computational simulators are suitable for reasonable larger PN models in comparison with the graphical ones. Moreover, Matlab itself makes possible to utilize prearranged computational procedures for the supervisor synthesis as well as to test the behaviour and properties of the final model of the supervised system.

The presented paper represents only Part 1 of the whole paper. It is planned to publish also the separate Part 2 of this paper where also several deeper case studies (including the error recovery approach at fault occurence, e.g. when a part drops from the robot gripper during an assembly process) will be presented in order to better document the applicability of the approaches on models of DES in a real environment (practice).

Acknowledgement

The author thanks for the partial support of the VEGA Agency (under Grant No. 2/0029/17).

REFERENCES

[1] Aguirre-Salas, L. I.—Santoyo-Sanchez, A.: Observability Analysis of Interpreted Petri Nets Under Partial State Observations Using Estimations Reachability Graph. Proceedings of the 2008 IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008), Hamburg, Germany, 2008, pp. 129–136, doi: 10.1109/ETFA.2008.4638383.

[2] Arichi, F.—Kebabti, H.—Cherki, B.—Djemai, M.: Failure Components Detection in Discrete Event Systems Modeled by Petri Net. Proceedins of IEEE 3rd International Conference on Systems and Control, October 2013, Algiers, Algeria, 2013, pp. 224–229, doi: 10.1109/ICoSC.2013.6750863.

[3] Cabasino, M. P.—Giua, A.—Seatzu, C.: Identification of Deterministic Petri Nets. Proceedings of the 8th International Workshop on Discrete Event Systems (WODES '06), Ann Arbor, Mich., USA, July 2006, pp. 325–331, doi: 10.1109/WODES.2006.382527.

[4] Cabasino, M. P.—Giua, A.—Seatzu, C.: Diagnosis Using Labeled Petri Nets with Silent or Undistinguishable Fault Events. IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems, Vol. 43, 2013, No. 2, pp. 345–355.

[5] Cabasino, M. P.—Hadjicostis, C. N.—Seatzu, C.: State Feedback Control of Labeled Petri Nets with Uncertainty in the Initial Marking. Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA 2014), Barcelona, Spain, September 2014, pp. 1–7, doi: 10.1109/ETFA.2014.7005154.

[6] Chao, D. Y.: Knitting Technique with TP-PT Generations for Petri Net Synthesis. Journal of Information Science and Engineering, Vol. 22, 2006, No. 4, pp. 909–923.

[7] Čapkovič, F.: Timed and Hybrid Petri Nets at Solving Problems of Computational Intelligence. Computing and Informatics, Vol. 34, 2015, No. 4, pp. 746–778.

[8] Čapkovič, F.: Interpreted Petri Nets in DES Control Synthesis. In: Nguyen, N. T., Trawiński, B., Fujita, H., Hong, T. P. (Eds.): Intelligent Information and Database

Systems (ACIIDS 2016). Springer, Berlin, Heidelberg, Lecture Notes in Artificial Intelligence, Vol. 9621, 2016, pp. 377–387, doi: 10.1007/978-3-662-49381-6_36.

[9] Čapkovič, F.: Petri Nets in Discrete-Event and Hybrid Systems Modelling, Analysing, Performance Evaluation and Control. In: Szewczyk, R., Zieliński, C. Kaliczyńska, M. (Eds.): Automation 2017 (ICA 2017). Innovations in Automation, Robotics and Measurement Techniques. Springer, Cham, Advances in Intelligent Systems and Computing, Vol. 550, 2017, pp. 3–21.

[10] Corona, D.—Giua, A.—Seatzu, C.: Marking Estimation of Petri Nets with Silent Transitions. Proceedings of the 43rd IEEE International Conference on Decision and Control (CDC), Atlantis, The Bahams, December 2004, pp. 966–971, doi: 10.1109/CDC.2004.1428810.

[11] Desel, J.—Reisig, W.: Place/Transition Petri Nets. In: Reisig, W., Rozenberg, G. (Eds.): Lectures on Petri Nets I: Basic Models. Advances in Petri Nets (ACPN 1996). Springer, Heidelberg, Lecture Notes in Computer Science, Vol. 1491, 1998, pp. 122–173.

[12] Desel, J.—Esparza, J.: Free Choice Petri Nets. Cambridge University Press, 1995.

[13] Elhog-Benzina, D.—Haddad, S.—Hennicker, R.: Refinement and Asynchronous Composition of Modal Petri Nets. In: Jensen, K., Donatelli, S., Kleijn, J. (Eds.): Transactions on Petri Nets and Other Models of Concurrency V. Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 6900, 2012, pp. 96–120.

[14] Esparza, J.—Nielsen, M.: Decidability Issues for Petri Nets. BRICS Report Series RS-94-8, May, 1994, 25 pp.

[15] Estrada-Vargas, A. P.—López-Mellado, E.—Lesage, J.-J.: A Black-Box Identification Method for Automated Discrete-Event Systems. IEEE Transactions on Automation Science and Engineering, IEEE, 2015, Vol. 14, No. 3, pp. 1321–1336, doi: 10.1109/TASE.2015.2445332.

[16] Estrada-Vargas, A. P.—López-Mellado, E.—Lesage, J.-J.: Identification of Partially Observable Discrete Event Manufacturing Systems. Proceedings of the 18th IEEE Conference on Emerging Technologies and Factory Automation (ETFA 2013), 2013, pp. 1–7, doi: 10.1109/ETFA.2013.6648052.

[17] Estrada-Vargas, A. P.—Lesage, J.-J.—López-Mellado, E.: A Stepwise Method for Identification of Controlled Discrete Manufacturing Systems. International Journal of Computer Integrated Manufacturing, Vol. 28, 2015, No. 2, pp. 187–199, doi: 10.1080/0951192X.2013.874591.

[18] Estrada-Vargas, A. P.—López-Mellado, E.—Lesage, J.-J.: A Comparative Analysis of Recent Identification Approaches for Discrete-Event Systems. Mathematical Problems in Engineering, Hindawi, Vol. 2010, 2010, Article ID 453254, 21 pp.

[19] Feng, L.—Wonham, W. M.: Supervisory Control Architecture for Discrete-Event Systems. IEEE Transactions on Automatic Control, Vol. 53, 2008, No. 6, pp. 1449–1461.

[20] Friedland, B.: Control System Design. McGraw Hill, 1986.

[21] Ghaffari, A.—Rezg, N.—Xie, X.: Design of a Live and Maximally Permissive Petri Net Controller Using the Theory of Regions. IEEE Transactions on Robotics and Automation, Vol. 19, 2003, No. 1, pp. 137–141, doi: 10.1109/TRA.2002.807555.

[22] Giua, A.—Seatzu, C.: Observability of Place/Transition Nets. IEEE Transactions on Automatic Control, Vol. 47, 2002, No. 9, pp. 1424–1437.

[23] Giua, A.—Corona, D.—Seatzu, C.: State Estimation of λ-free Labeled Petri Nets with Contact-Free Non-Deterministic Transitions. Discrete Event Dynamic Systems: Theory and Applications, Vol. 15, No. 1, 2005, pp. 85–108, doi: 10.1007/s10626-005-5239-4.

[24] Giua, A.—Seatzu, C.: A System Theory View of Petri Nets. In: Bonivento, C., Marconi, L., Rossi, C., Isidori, A. (Eds.): Advances in Control Theory and Applications. Springer, Berlin, Heidelberg, Lecture Notes in Control and Information Sciences, Vol. 353, 2007, pp. 99–127.

[25] Giua, A.—Seatzu, C.: Observability Properties of Petri Nets. Proceedings 39th IEEE Conference on Decision and Control, Sydney, Australia, Vol. 3, 2000, pp. 2676–2681, doi: 10.1109/CDC.2000.914209.

[26] Giua, A.—Seatzu, C.: Identification of Free-Labeled Petri Nets via Integer Programming. Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005, Seville, Spain, December 12–15, 2005, pp. 7639–7644.

[27] Hiraishi, K.—Čapkovič, F.: Construction of Rule Base for the Control of Discrete Event Dynamic Systems. Research Report IS-RR-2002-001, ISSN 0918-7553, Japan Advanced Institute of Science and Technology (JAIST), Kanazawa, Japan, 2002, pp. 1–8.

[28] Holloway, L.E.—Khare, A.S.—Gong, Y.: Computing Bounds for Forbidden State Reachability Functions for Controlled Petri Nets. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, Vol. 34, No. 2, March 2004, pp. 219–228, doi: 10.1109/TSMCA.2003.822279.

[29] Holloway, L.E.—Krogh, B.H.—Giua, A.: A Survey of Petri Net Methods for Controlled Discrete Event Systems. Discrete Event Dynamic Systems: Theory and Applications, Vol. 7, 1997, No. 2, pp. 151–190.

[30] Holloway, L.E.—Krogh, B.H.: Controlled Petri Nets: A Tutorial Survey. In: Cohen, G., Quadrat, J. P. (Eds.): Proceedings of 11th International Conference on Analysis and Optimization of Systems Discrete Event Systems. Springer, Berlin, Heidelberg, Lecture Notes in Control and Information Sciences, Vol. 199, 1994, pp. 158–168, doi: 10.1007/BFb0033544.

[31] Holloway, L. E.—Guan, X.—Zhang, L.: A Generalization of State Avoidance Policies for Controlled Petri Nets. IEEE Transactions on Automatic Control, Vol. 41, 1996, No. 6, pp. 804–816, doi: 10.1109/9.506233.

[32] Ichikawa, A.—Hiraishi, K.: Analysis and Control of Discrete Event Systems Represented by Petri Nets. In: Varaiya, P., Kurzhanski, A. B. (Eds.): Discrete Event Systems: Models and Applications. Springer, Berlin, Heidelberg, Lecture Notes in Control and Information Sciences, Vol. 103, 1988, pp. 115–134.

[33] Iordache, M.—Antsaklis, P.: Supervisory Control of Concurrent Systems: A Petri Net Structural Approach. Birkhäuser, 2006.

[34] Iordache, M.—Antsaklis, P.: Supervision Based on Place Invariants: A Survey. Discrete Event Dynamic Systems, Vol. 16, 2006, No. 4, pp. 451–492, doi: 10.1007/s10626-006-0021-9.

[35] Jantzen M.: Language Theory of Petri Nets. In: Brauer, W., Reisig, W., Rozenberg, G. (Eds.): Petri Nets: Central Models and Their Properties (ACPN 1986). Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 254, 1987, pp. 397–412, doi: 10.1007/978-3-540-47919-2_15.

[36] Krogh, B. H.: Controlled Petri Nets and Maximally Permissive Feedback Logic. Proceedings 25th Annual Allerton Conference on Communication Control, and Computing, University of Illinois, Urbana, 1987, pp. 317–326.

[37] Kunimochi, Y.: Algebraic Properties of Petri Net Languages and Codes. Ph.D. Thesis, University of Debrecen, Faculty of Informatics, Debrecen, Hungary, 2009, 89 pp.

[38] Kwakernaak, H.—Sivan, R.: Linear Optimal Control Systems. John Wiley & Sons Inc. (Wiley-Interscience), New York-Chichester-Brisbane-Toronto, 1972.

[39] Lutz-Ley, A.—López Mellado, E.: Synthesis of Fault Recovery Sequences in a Class of Controlled Discrete Event Systems Modelled with Petri Nets. Procedia Technology, Vol. 7, 2013, pp. 257–264, doi: 10.1016/j.protcy.2013.04.032.

[40] Mandrioli, D.: A Note on Petri Net Languages. Information and Control, Vol. 34, 1977, No. 2, pp. 169–171, doi: 10.1016/S0019-9958(77)80013-2.

[41] Meda-Campaña, M. E.—López-Mellado, E.: Incremental Synthesis of Petri Net Models for Identification of Discrete Event Systems. Proceedings 41st IEEE Conference on Decision and Control, Las Vegas, Nevada, USA, December 2002, pp. 805–810.

[42] Meda-Campaña, M. E.—López-Mellado, E.: Required Event Sequences for Identification of Discrete Event Systems. Proceedings 42nd IEEE Conference on Decision and Control, Maui, Hawaii, USA, December 2003, pp. 3778–3783.

[43] Moody, J. O.—Antsaklis, P. J.: Petri Net Supervisors for DES with Uncontrollable and Unobservable Transitions. IEEE Transactions on Automatic Control, Vol. 45, No. 3, March 2000, pp. 462–476, doi: 10.1109/9.847725.

[44] Moody, J. O.—Antsaklis, P. J.: Uncontrollable and Unobservable Transitions. In: Moody, J. O., Antsaklis, P. J. (Eds.): Supervisory Control of Discrete Event Systems Using Petri Nets. Chapter 4. Springer, Boston, The International Series on Discrete Event Dynamic Systems, Vol. 8, 1998, pp. 33–50.

[45] Yamalidou, K.—Moody, J.—Lemmon, M. D.—Antsaklis, P. J.: Feedback Control of Petri Nets Based on Place Invariants. Proceedings of the 33rd IEEE Conference on Decision and Control, Lake Buena Vista, FL, December 14–16, 1994, pp. 3104–3109.

[46] Yamalidou, K.—Moody, J.—Lemmon, M. D.—Antsaklis, P. J.: Feedback Control of Petri Nets Based on Place Invariants. Automatica, Vol. 32, No. 1, 1996, pp. 15–28, doi: 10.1016/0005-1098(95)00103-4.

[47] Murata, T.: Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE, Vol. 77, 1989, No. 4. pp. 541–580, doi: 10.1109/5.24143.

[48] Ogata, K.: Modern Control Engineering. 3rd Edition, Prentice Hall, 1999.

[49] Peterson, J. L.: Petri Net Theory and the Modeling of Systems. Prentice-Hall, Englewood Cliffs, NJ, 1981.

[50] Ramadge, J. G.— Wonham, W. M.: The Control of Discrete Event Systems. Proceedings of the IEEE, Vol. 77, 1989, No. 1, pp. 81–98, doi: 10.1109/5.21072.

[51] Ramírez—Treviño, A.—Rivera-Rangel, I.—López-Mellado, E.: Observer Design for Discrete Event Systems Modeled by Interpreted Petri Nets. IEEE International Conference on Robotics and Automation (ICRA), 2000.

[52] Ramírez—Treviño, A.—Rivera-Rangel, I.—López-Mellado, E.: Observability of Discrete Event Systems Modeled by Interpreted Petri Nets. IEEE Transactions on Robotics and Automation, Vol. 19, No. 4, August 2003, pp. 557–565.

[53] Ramírez—Treviño, A.—Ruiz–Beltrán, E.—Rivera-Rangel, I.—López-Mellado, E.: Online Fault Diagnosis of Discrete Event Systems. A Petri Net-Based Approach. IEEE Transactions on Automation Science and Engineering, Vol. 4, 2007, No. 1, pp. 31–39.

[54] Rivera-Rangel, I.—Ramírez-Treviño, A.—Aguirre-Salas, L. I.—Ruiz-Léon, J.: Geometrical Characterization of Observability in Interpreted Petri Nets. Kybernetika, Vol. 41, 2005, No. 5, pp. 553–574.

[55] Ru, Y.—Hadjicostis, C. N.: State Estimation in Discrete Event Systems Modeled by Labeled Petri Nets. Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, CA, USA, 2006, pp. 6022–6027, doi: 10.1109/CDC.2006.377126.

[56] Ru, Y.—Hadjicostis, C. N.: Sensor Selection for Structural Observability in Discrete Event Systems Modeled by Petri Nets. IEEE Transactions on Automatic Control, Vol. 55, 2010, No. 8, pp. 1751–1764.

[57] Santoyo-Sanchez, A.—Pérez-Martínez, M. A.—De Jesús-Velásquez, C.—Aguirre-Salas, L. I.—Alvarez-Ureña, M. A.: Modeling Methodology for NPC's Using Interpreted Petri Nets and Feedback Control. Proceedings of the 2010 7th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE 2010), Tuxtla Gutiérrez, Chiapas, Mexico, September 8–10, 2010, IEEE Press, 2010, pp. 369–374.

[58] Santoyo-Sanchez, A.—Ramírez-Treviño, A.—De Jesús-Velásquez, C.—Aguirre-Salas, L. I.: Step State-Feedback Supervisory Control of Discrete Event Systems Using Interpreted Petri Nets. Proceedings of the 2008 IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008), Hamburg, Germany, 2008, pp. 926–933.

[59] Sessego, F.—Giua, A.—Seatzu, C.: HYPENS: A Matlab Tool for Timed Discrete, Continuous and Hybrid Petri Nets. In: van Hee, K. M., Valk, R. (Eds.): Applications and Theory of Petri Nets (PETRI NETS 2008). Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 5062, 2008, pp. 419–428.

[60] Tapia-Flores, T.—López-Mellado, E.—Estrada-Vargas, A. P.—Lesage, J.-J.: Petri Net Discovery of Discrete Event Processes by Computing T-Invariants. Proceedings of 19th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA '14), Barcelona, Spain, 2014, Art. No. 345, 8 pp.

[61] THISTLE, J. G.: Supervisory Control of Discrete Event Systems. Mathematical and Computer Modelling, Vol. 23, 1996, No. 11-12, pp. 25–53, doi: 10.1016/0895-7177(96)00063-5.

[62] VAN GLABBEEK, R. J.: The Individual and Collective Token Interpretations of Petri Nets. In: Abadi, M., de Alfaro, L. (Eds.): Concurrency Theory (CONCUR 2005). Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 3653, 2005, pp. 323–337, doi: 10.1007/11539452_26.

[63] VALK, R.—VIDAL-NAQUET, G.: Petri Nets and Regular Languages. Journal of Computer and System Sciences, Vol. 23, 1981, No. 3, pp. 299–325.

[64] WANG, J.: Timed Petri Nets. Kluwer Academic Publishers, Boston, MA, USA, 1998, doi: 10.1007/978-1-4615-5537-7.

[65] ZHANG, L.—HOLLOWAY, L. E.: Forbidden State Avoidance in Controlled Petri Nets under Partial Observation. Proceedings of 33rd Allerton Conference, Monticello, Illinois, 1995, pp. 146–155.

[66] ZHAO, M.—UZAM, M.—HOU, Y. F.: Near-Optimal Supervisory Control of Flexible Manufacturing Systems Using Divide-and-Conquer Iterative Method. Advances in Mechanical Engineering, Vol. 8, 2016, No. 3, pp. 1–17.

František ČAPKOVIČ received his Master degree in 1972 from the Faculty of Electrical Engineering of the Slovak Technical University, Bratislava, Slovakia. Since 1972 he has been working at the Slovak Academy of Sciences (SAS), Bratislava, namely in 1972–1991 at the Institute of Technical Cybernetics, in 1991–2001 at the Institute of Control Theory and Robotics and in 2001 till now at the Institute of Informatics. In 1980 he received his Ph.D. from SAS. In 1998 he was appointed the Associate Professor. He works in the area of modelling, analysing and intelligent control of Discrete-Event Systems (DES) and Hybrid Systems. He is the author of more than 240 publications.