Modularity and Supervision in Agents Cooperation

(Compendium of Recent Research)

František Čapkovič

Institute of Informatics, Slovak Academy of Sciences

Dúbravská cesta 9, 845 07 Bratislava, Slovakia

E-mail: Frantisek.Capkovic@savba.sk

URL: http://www.ui.sav.sk/home/capkovic/capkhome.htm

Kyoto University, JAPAN

May 10-June 7, 2011

Content

Discrete Event Systems (DES) are systems discrete in nature - i.e. driven by discrete events. Namely, the course of a DES variable evolves in response to certain discrete qualitative changes, called events:

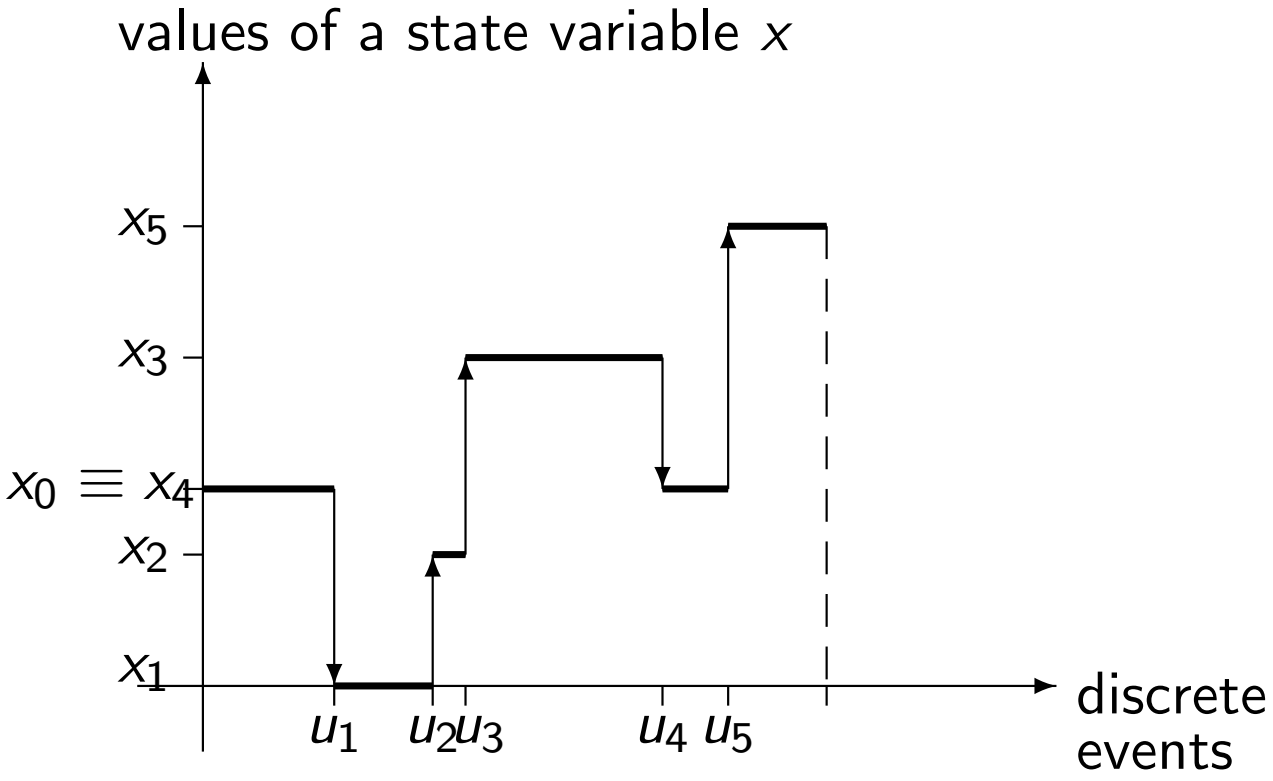

Figure 1. *The development of a state variable x of DES.*

Formally, DES can be described by the triplet

$$\langle \mathcal{X}, \mathcal{U}, \mathcal{A} \rangle$$

where

$\mathcal{X} = \{X_0, X_1, ..., X_N\}$ is the set of the DES states (the state space)
$X_k = \{{}^k x_1, .., {}^k x_n\}$, $k = 0, 1, ..., N$, ${}^k x_i \in \{0, 1, ..., {}^k c_i\}$, $i = 1, ..., n$
is the set of the states of DES elementary (atomic) subsystems

$\mathcal{U} = \{U_0, U_1, ..., U_{N-1}\}$ is the set of the DES discrete events
$U_k = \{{}^k u_1, .., {}^k u_m\}$, $k = 0, 1, ..., N-1$, ${}^k u_j \in \{0, 1\}$, $j = 1, ..., m$
is the set of DES elementary (atomic) discrete events

$\mathcal{A} \subseteq \mathcal{X} \times \mathcal{U} \rightarrow \mathcal{X}$ is the set of mutual causal relations among the states and the discrete events.

- Agents are usually understood to be persistent (software, but not only software) entities that can perceive, reason, and act in their environment and communicate with other agents.
- From the external point of view the agent (a real or virtual entity)
  - evolves in an environment
  - is able to perceive this environment
  - is able to act in this environment
  - is able to communicate with other agents
  - exhibits an autonomous behaviour
- From the internal point of view the agent
  - encompasses some local control in some of its perception, communication, knowledge acquisition, reasoning, decision, execution, and action processes

Because agents and groups of agents exhibit the behaviour driven by discrete events, DES can be utilized for modelling agents and multi agent systems (MAS).

1. Place/Transition Petri nets (P/T PN) as a description tool
   - PN defined in [Peterson, 1981; Murata, 1989] are widely used to describe the behaviour of DES
   - PN are bipartite directed graphs with two kinds of nodes and two kinds of edges.
   - Agents and Multi Agent Systems (MAS) can be (observing their behaviour) understood to be a kind of DES.
2. Why Petri Nets?
   - PN yields both the graphical model and the mathematical one
   - PN have a formal semantics
   - There are many techniques for proving PN basic properties like reachability, liveness, boundedness, conservativeness, reversibility, coverability, persistence, fairness, etc.
   - Consequently, PN represent the enough general means to be able to model a wide class of systems, including agents and MAS.

As to PN structure, formally they are the quadruplet

$$\langle \mathcal{P}, \mathcal{T}, \mathcal{F}, \mathcal{G} \rangle; \qquad \mathcal{P} \cap \mathcal{T} = \emptyset, \qquad \mathcal{F} \cap \mathcal{G} = \emptyset$$

where

$\mathcal{P} = \{p_1, p_2, ..., p_n\}$ is set of PN places; $p_i \in \mathcal{P}, \ i = 1, ..., n$ represents the states of elementary (atomic) activities of DES

$\mathcal{T} = \{t_1, t_2, ..., t_m\}$ is set of PN transitions; $t_j \in \mathcal{T}, \ j = 1, ..., m$ represents the discrete events

$\mathcal{F} \subseteq \mathcal{P} \times \mathcal{T}$ is the set of interconnections (causal relations) from places to transitions $(\mathcal{P} \to \mathcal{T})$

$\mathcal{G} \subseteq \mathcal{T} \times \mathcal{P}$ is the set of interconnections (causal relations) from transitions to places $(\mathcal{T} \to \mathcal{P})$

As to PN dynamics, formally they are the quadruplet

$$\langle \mathcal{X}, \mathcal{U}, \delta, \mathbf{x}_0 \rangle$$

where

$\mathcal{X} = \{\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_N\}$ is the set of state vectors;
$\mathbf{x}_k \in \mathcal{X}, \ k = 0, 1, ..., N$ represents the state vectors of elementary places;
$\mathbf{x}_k = (\sigma^k_{p_1}, ..., \sigma^k_{p_n})^T; \quad \sigma^k_{p_i} \in \{0, 1, ..., c_{p_i}\}, \ i = 1, ..., n; \quad 0 \leq \sigma^k_{p_i} \leq c_{p_i};$
c_{p_i} is the capacity

$\mathcal{U} = \{\mathbf{u}_0, \mathbf{u}_1, ..., \mathbf{u}_M\}$ is the set of control vectors;
$\mathbf{u}_k \in \mathcal{U}, \ k = 0, 1, ..., M$ represents the state vectors of elementary transitions; $\mathbf{u}_k = (\gamma^k_{t_1}, ..., \gamma^k_{t_m})^T; \quad \gamma^k_{t_j} \in \{0, 1\}, \ j = 1, ..., m;$

$\delta : \mathcal{X} \times \mathcal{U} \to \mathcal{X}$ is the transition function

$\mathbf{x}_0$ is the initial state vector

The following linear discrete system is the effective PN model

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + \mathbf{B}.\mathbf{u}_k \quad , \quad k = 0, \ldots, K \\ \mathbf{B} &= \mathbf{G}^T - \mathbf{F} \\ \mathbf{F}.\mathbf{u}_k &\leq \mathbf{x}_k \end{aligned}$$

where

k is the discrete step of the dynamics development

$\mathbf{B}$, $\mathbf{F}$, $\mathbf{G}$ are structural matrices of constant elements

$\mathbf{F} = \{f_{ij}\}_{n \times m}$, $f_{ij} \in \{0, M_{f_{ij}}\}$, $i = 1, \ldots, n$, $j = 1, \ldots, m$ express the causal relations between the states and the discrete events

$\mathbf{G} = \{g_{ij}\}_{m \times n}$, where $g_{ij} \in \{0, M_{g_{ij}}\}$, $i = 1, \ldots, m$, $j = 1, \ldots, n$ express the causal relations between the discrete events and the DEDS states

Example of a PN:

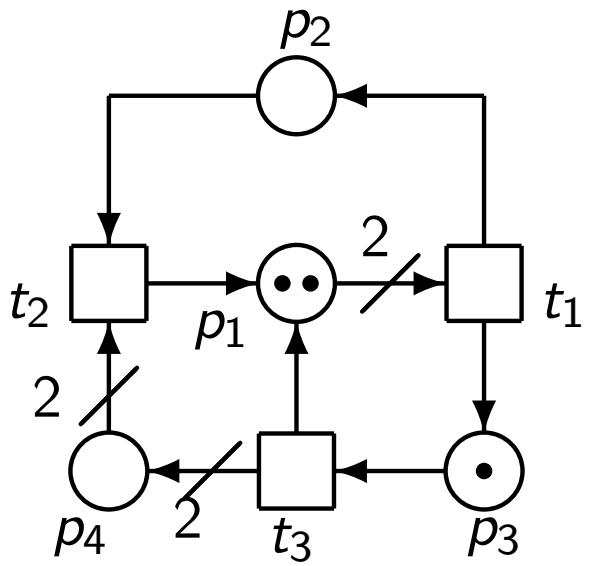

Figure 2. The PN-based model of the DEDS subsystem

$$\mathcal{P} = \{p_1, \ldots, p_4\}, \mathcal{T} = \{t_1, \ldots, t_3\},$$

$$\mathcal{F} \to \mathbf{F} = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 2 & 0 \end{pmatrix}, \mathcal{G} \to \mathbf{G} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 \end{pmatrix}, \mathbf{x}_0 = \begin{pmatrix} 2 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

The enabled transition and its firing:

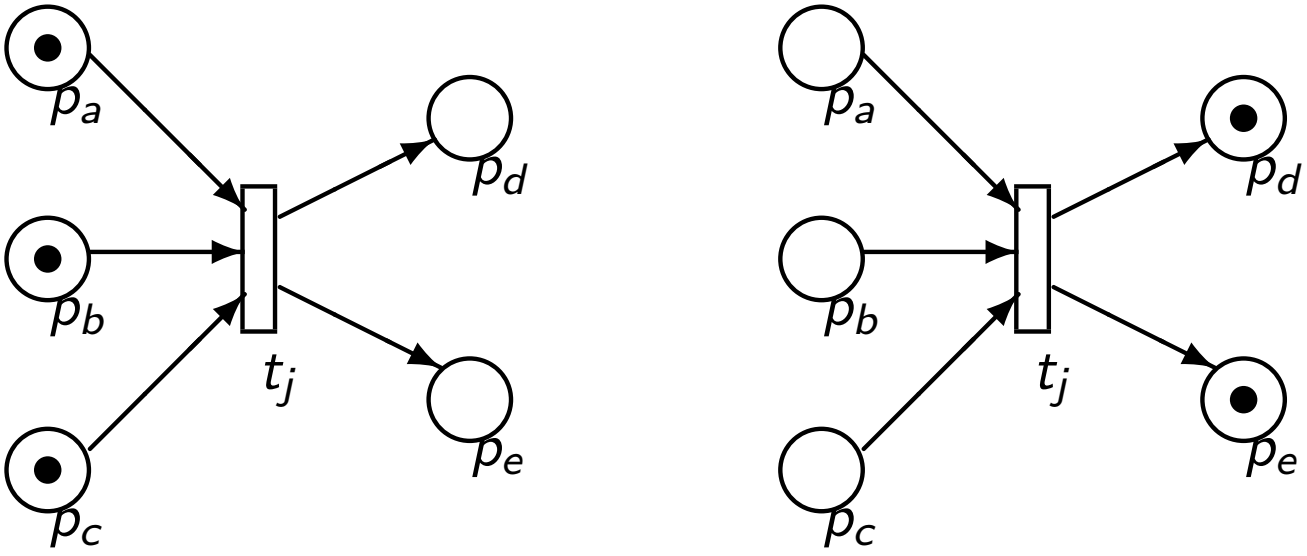

Figure 3. The enabled transition t_j (the left picture) and the state after its firing (the right picture).

Which PN transitions are enabled? Namely, enabled PN transitions can be fired (but not simultaneously) and change the PN state in such a way.

$$\begin{aligned}
\mathbf{u}_k &= \text{neg}(\mathbf{F}^T.\text{neg}(\mathbf{x}_k)) \\
\mathbf{u}_0 &= \text{neg}(\mathbf{F}^T.\text{neg}(\mathbf{x}_0)) = (1,\ 0,\ 1)^T \\
\textbf{Because} & \quad \text{only one of the enabled transitions can be fired} \\
\mathbf{u}_0 &= (1,\ 0,\ 0)^T \quad \text{or} \quad \mathbf{u}_0 = (0,\ 0,\ 1)^T \\
\mathbf{x}_1 &= \mathbf{x}_0 + \mathbf{B}.\mathbf{u}_0
\end{aligned}$$

$$\text{if} \quad \mathbf{u}_0 = \begin{cases} (1,\ 0,\ 0)^T & \text{then} \quad \mathbf{x}_1 = (0,\ 1,\ 2,\ 0)^T \\ (0,\ 0,\ 1)^T & \text{then} \quad \mathbf{x}_1 = (3,\ 0,\ 0,\ 2)^T \end{cases}$$

$$\text{if} \quad \mathbf{x}_1 = \begin{cases} (0,\ 1,\ 2,\ 0)^T & \text{then} \quad \mathbf{u}_1 = (0,\ 1,\ 1)^T \\ (3,\ 0,\ 0,\ 2)^T & \text{then} \quad \mathbf{u}_1 = (1,\ 0,\ 0)^T \end{cases}$$

It means that a branching occurs $\Rightarrow$ Reachability Tree of the PN.

The reachability tree (RT) and the reachability graph (RG) corresponding to the PN.

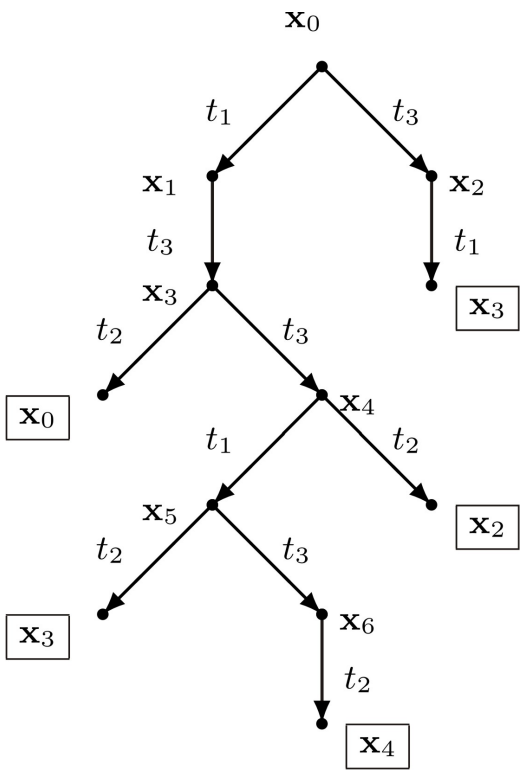

Figure 4.a. *The reachability tree of the PN*

The RG corresponding to the RT can be found by connecting the RT nodes with the same name.

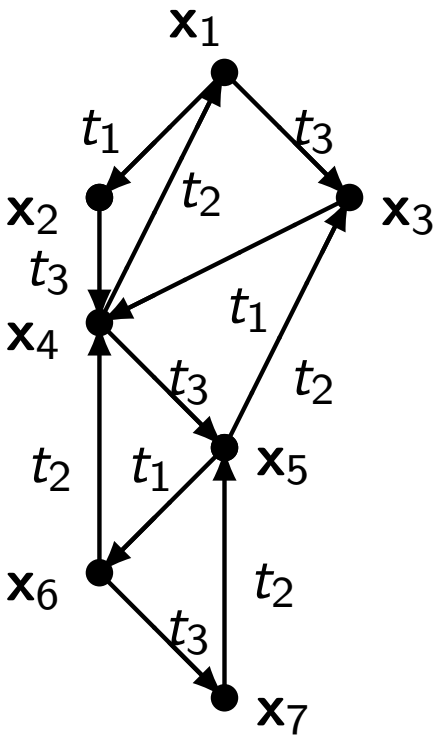

Figure 4.b. *The reachability graph of the PN*

The RT adjacency matrix is

$$\mathbf{A}_{RT} = \begin{pmatrix} 0 & 1 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 \end{pmatrix}$$

The set of the reachable states can be expressed by columns of

$$\mathbf{X}_{reach} = \begin{pmatrix} 2 & 0 & 3 & 1 & 2 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 2 & 2 \\ 1 & 2 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 2 & 2 & 4 & 4 & 6 \end{pmatrix}$$

The PN simulator

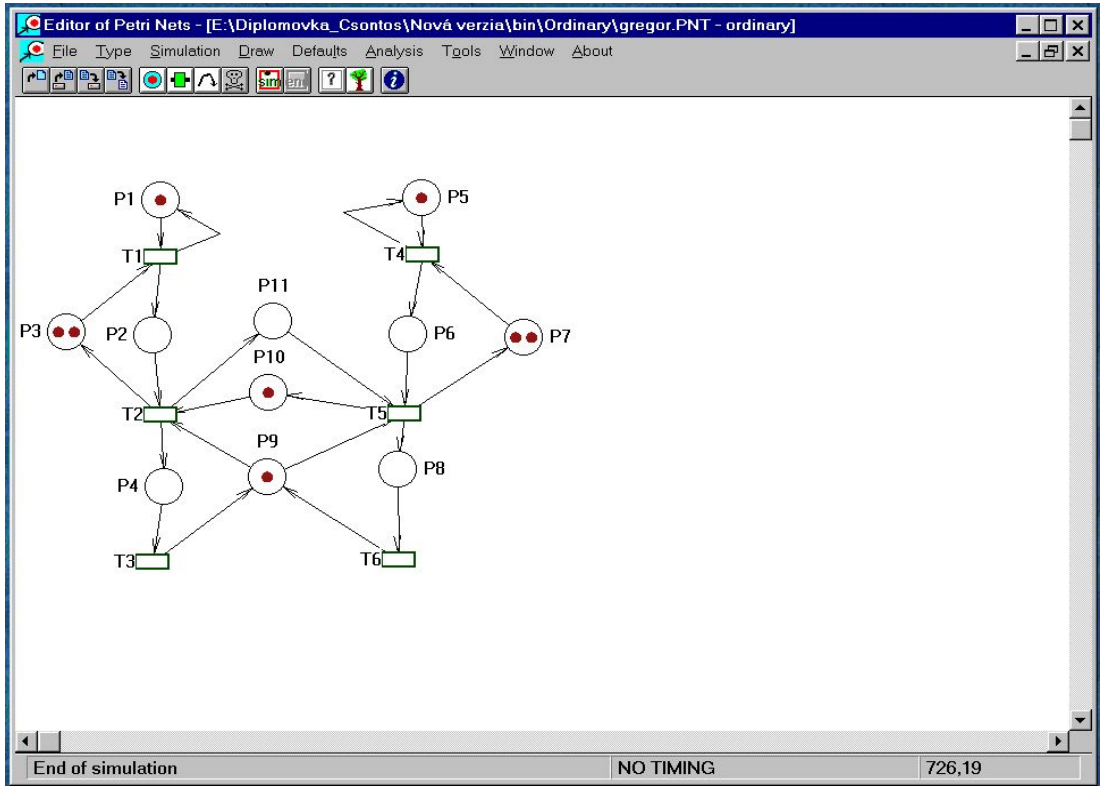

Figure 5. *The simulator screen at simulation of a PN*

The RT of the PN displayed by the simulator

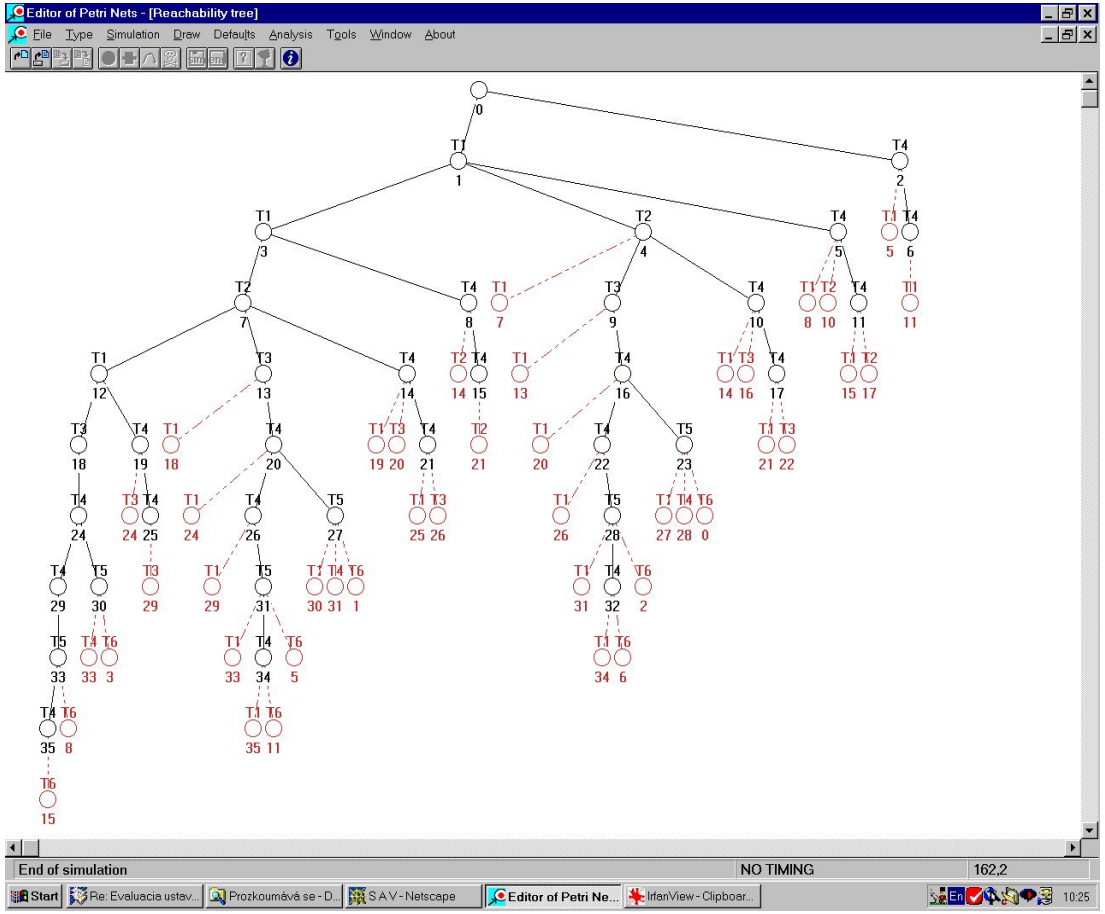

Figure 6. *The simulator screen with the PN reachability tree*

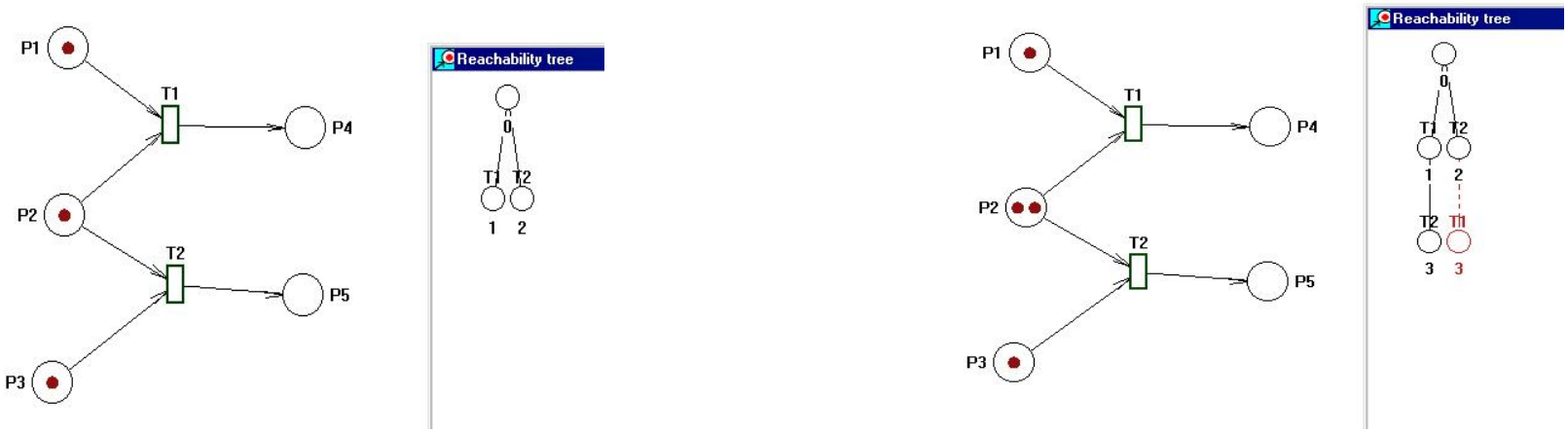

Figure 7. *The illustration of the PN modelling the conflict (the left picture) and parallelism (the right picture)*

The RG of a PN displayed by the RG simulator GraSim

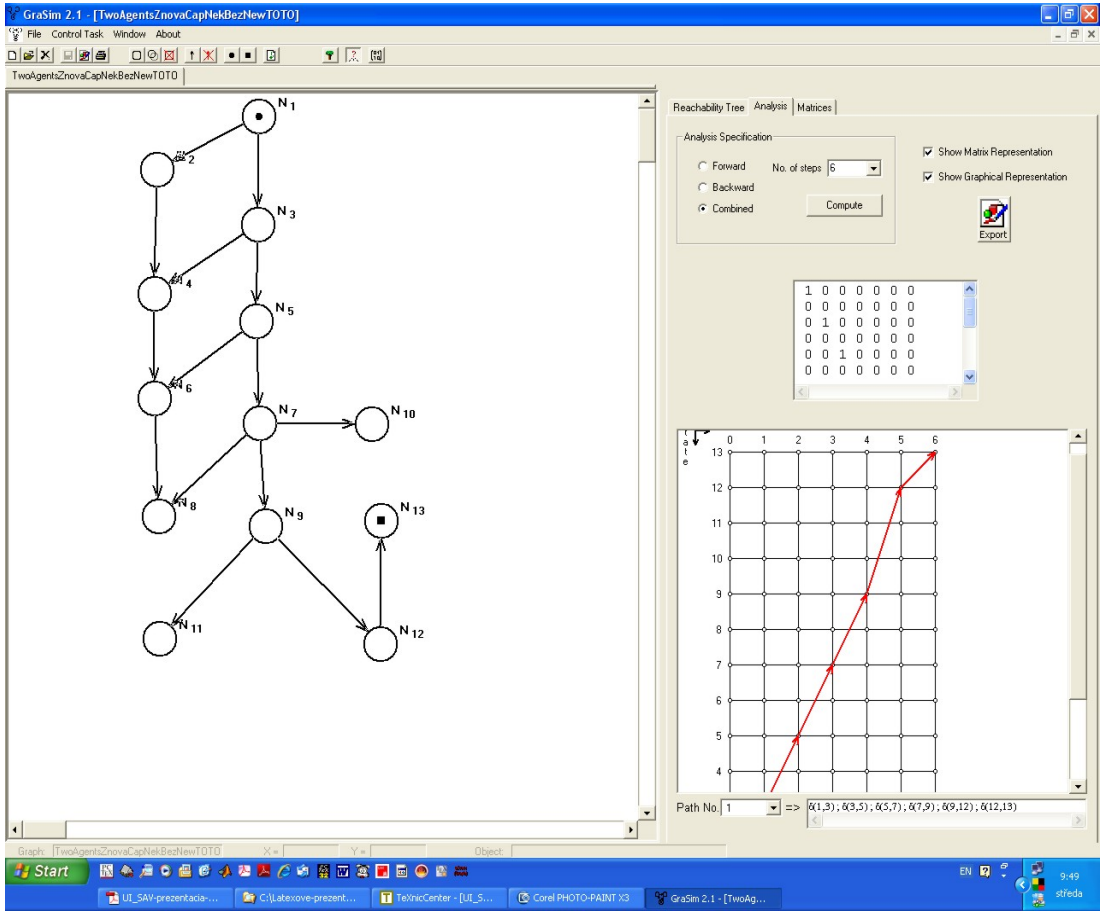

Figure 8. *The RG simulator screen with the PN reachability graph*

2. Modularity in DES

The structure of a group of autonomous DES modules (agents) can be expressed by means of diagonal incidence matrices

$$\mathbf{F} = \begin{pmatrix} \mathbf{F}_1 & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_2 & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{F}_{N_A-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{F}_{N_A} \end{pmatrix} = \text{blockdiag}(\mathbf{F}_i)_{i=1,N_A}$$

$$\mathbf{G} = \begin{pmatrix} \mathbf{G}_1 & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{G}_{N_A-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{G}_{N_A} \end{pmatrix} = \text{blockdiag}(\mathbf{G}_i)_{i=1,N_A}$$

There are three possibilities how to connect the DES modules (agents):

1. by PN transitions - here the DES modules modelled by PN subnets are mutually connected by the interface consisting from PN transitions
2. by PN places - here the DES modules modelled by PN subnets are mutually connected by the interface consisting from PN places
3. by both the PN transitions and the PN places - here the DES modules modelled by PN subnets are mutually connected by the interface consisting from PN transitions and PN places, i.e. by a PN subnet

Interconnections by PN transitions

$$\mathbf{F}=\left(\begin{array}{ccccc|c} \mathbf{F}_1 & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{F}_{c_1} \\ \mathbf{0} & \mathbf{F}_2 & \dots & \mathbf{0} & \mathbf{0} & \mathbf{F}_{c_2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{F}_{N_A-1} & \mathbf{0} & \mathbf{F}_{c_{N_A-1}} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{F}_{N_A} & \mathbf{F}_{c_{N_A}} \end{array}\right)$$

$$\mathbf{G}=\left(\begin{array}{ccccc} \mathbf{G}_1 & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{G}_{N_A-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{G}_{N_A} \\ \hline \mathbf{G}_{c_1} & \mathbf{G}_{c_2} & \dots & \mathbf{G}_{c_{N_A-1}} & \mathbf{G}_{c_{N_A}} \end{array}\right)$$

$$\mathbf{F} = \left(\ \text{blockdiag}(\mathbf{F}_i)_{i=1,N_A} \ \ | \mathbf{F}_c \ \right)$$

$$\mathbf{G} = \left(\begin{array}{c} \text{blockdiag}(\mathbf{G}_i)_{i=1,N_A} \\ \text{----------} \\ \mathbf{G}_c \end{array} \right)$$

$$\mathbf{B} = \left(\ \text{blockdiag}(\mathbf{B}_i)_{i=1,N_A} \ \ | \mathbf{B}_c \ \right)$$

where $\mathbf{B}_i = \mathbf{G}_i^T - \mathbf{F}_i$; $\mathbf{B}_{c_i} = \mathbf{G}_{c_i}^T - \mathbf{F}_{c_i}$; $i = 1, \ldots, N_A$;
$\mathbf{F}_c = (\mathbf{F}_{c_1}^T, \mathbf{F}_{c_2}^T, \ldots, \mathbf{F}_{c_{N_A}}^T)^T$; $\mathbf{G}_c = (\mathbf{G}_{c_1}, \mathbf{G}_{c_2}, \ldots, \mathbf{G}_{c_{N_A}})$
$\mathbf{B}_c = (\mathbf{B}_{c_1}^T, \mathbf{B}_{c_2}^T, \ldots, \mathbf{B}_{c_{N_A}}^T)^T$.

Here, $\mathbf{F}_i$, $\mathbf{G}_i$, $\mathbf{B}_i$ represent the parameters of the PN-based model of A_i.
$\mathbf{F}_c$, $\mathbf{G}_c$, $\mathbf{B}_c$ represent the structure of the interface between the agents.
This interface consists (exclusively) of additional PN transitions.

Communication of Agent Models through the PN Transitions

Consider three agents (e.g. intelligent robots) A_1, A_2, A_3.

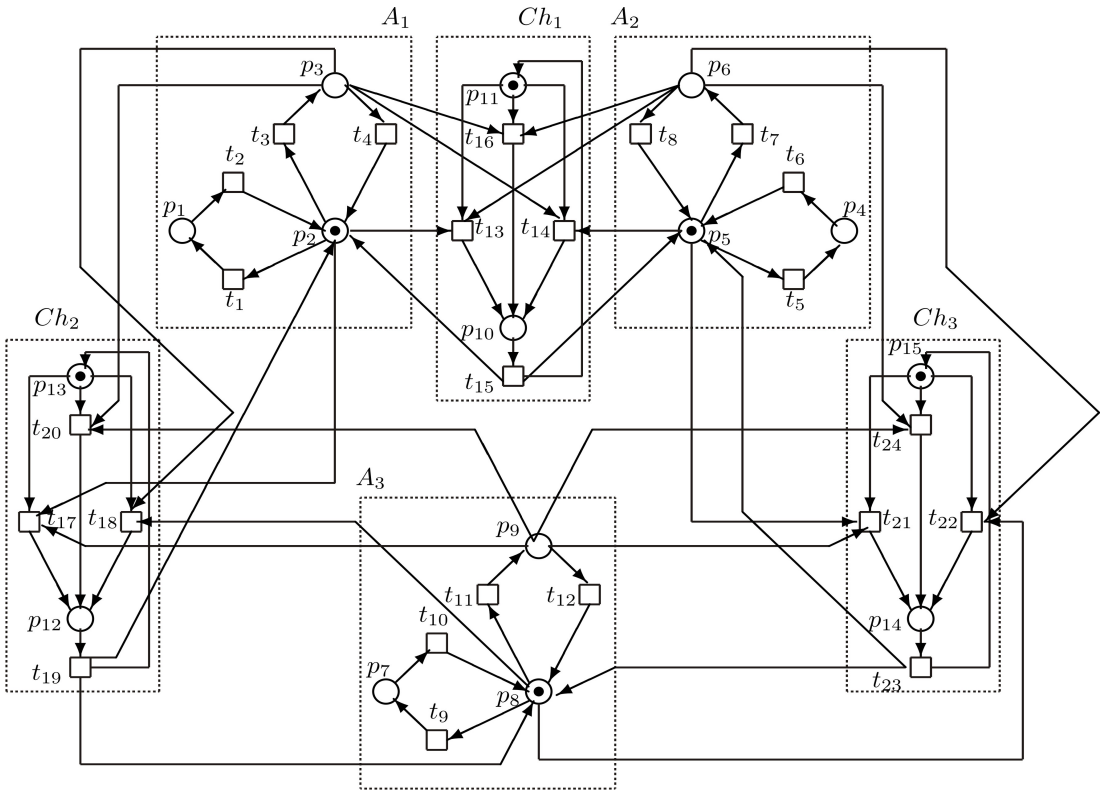

Figure 9. *The communication of three agents A_1, A_2, A_3*

The sets of the places of the agents PN models are
$\mathcal{P}_{A_1} = \{p_1, p_2, p_3\}$, $\mathcal{P}_{A_2} = \{p_4, p_5, p_6\}$, $\mathcal{P}_{A_3} = \{p_7, p_8, p_9\}$,
while the sets of transitions of their PN models are
$\mathcal{T}_{A_1} = \{t_1, t_2, t_3, t_4\}$, $\mathcal{T}_{A_2} = \{t_5, t_6, t_7, t_8\}$, $\mathcal{T}_{A_3} = \{t_9, t_{10}, t_{11}, t_{12}\}$.

The places represents three basic states of the agents
- the particular agent is either available (p_2, p_5, p_8) or
- it wants to communicate (p_3, p_6, p_9) or
- it does not want to communicate (p_1, p_4, p_7).

The autonomous agents have the same structure given as follows

$$\mathbf{F}_{A_i} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}; \ \mathbf{G}^T_{A_i} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}; \ i = 1, 2, 3$$

The communication channels between the corresponding two agents:

Ch_1 between A_1 and A_2 consists of $\{p_{10}, p_{11}\}$, $\{t_{13}, t_{14}, t_{15}, t_{16}\}$
Ch_2 between A_1 and A_3 consists of $\{p_{12}, p_{13}\}$, $\{t_{17}, t_{18}, t_{19}, t_{20}\}$
Ch_3 between A_2 and A_3 consists of $\{p_{14}, p_{15}\}$, $\{t_{21}, t_{22}, t_{23}, t_{24}\}$.

The states of the channels are:

- available (p_{11}, p_{13}, p_{15})
- realizing the communication of corresponding agents (p_{10}, p_{12}, p_{14}).

The channels create the interface between the communicating agents.
They can also be understood to be the agents.
The structure of communication channels between the particular agents is

$$\mathbf{F}_{Ch_i} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}; \ \mathbf{G}^T_{Ch_i} = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}; \ i = 1, 2, 3$$

$$\mathbf{F}_c = \left(\begin{array}{cccc|cccc|cccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1
\end{array}\right)$$

$$\mathbf{G}_c^T = \left(\begin{array}{cccc|cccc|cccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right)$$

$$\mathbf{F}_A = \begin{pmatrix} \mathbf{F}_{A_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{A_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_{A_3} \end{pmatrix}; \ \mathbf{G}_A^T = \begin{pmatrix} \mathbf{G}_{A_1}^T & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{A_2}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_{A_3}^T \end{pmatrix}$$

$$\mathbf{F}_{Ch} = \begin{pmatrix} \mathbf{F}_{Ch_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{Ch_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_{Ch_3} \end{pmatrix}; \ \mathbf{G}_{Ch}^T = \begin{pmatrix} \mathbf{G}_{Ch_1}^T & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{Ch_2}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_{Ch_3}^T \end{pmatrix}$$

$$\mathbf{F} = \begin{pmatrix} \mathbf{F}_A & \mathbf{F}_c \\ \mathbf{0} & \mathbf{F}_{Ch} \end{pmatrix}; \ \mathbf{G}^T = \begin{pmatrix} \mathbf{G}_A^T & \mathbf{G}_c^T \\ \mathbf{0} & \mathbf{G}_{Ch}^T \end{pmatrix}$$

Starting from the initial state
$\mathbf{x}_0 = (\mathbf{x}_{A_1}^T 0,\ \mathbf{x}_{A_2}^T 0,\ \mathbf{x}_{A_3}^T 0,\ \mathbf{x}_{Ch_1}^T 0,\ \mathbf{x}_{Ch_2}^T 0,\ \mathbf{x}_{Ch_3}^T 0)^T$, where $\mathbf{x}_{A_i}^T 0 = (0,\ 1,\ 0)^T$, $\mathbf{x}_{Ch_i}^T 0 = (0,\ 1)^T$, $i = 1,\ 2,\ 3$
we obtain the reachability graph with 36 nodes.
It represents the space of feasible states reachable form the initial state $\mathbf{x}_0$.

The feasible states are given as the columns of the matrix

$$\mathbf{x}_{reach} =$$

$$\begin{pmatrix}
0&1&0&0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
1&0&0&1&1&1&1&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&1&1&0&1&0&0\\
0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&1&1&0&0&0&1&1&0&0&0&1&1&1&0&0&0&0&0&0\\
1&1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&0&1&1&0&0&0&1&1&0&0&0&0&1&1&1&0&0&0\\
0&0&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&1&0&0&1&0&1&0&0&1&0&0&1&0&0&0&1&0\\
1&1&1&1&1&0&0&1&1&0&0&1&1&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&1&0&0&1&0&1&0&0&1&0&0&1&0&0&0&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&0&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&0&0\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&0&1&1&1&1&1&1&1&1&1&1&1&1&0&1&1&0&1&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&1&0&0&0&0&0&0&1&0&0\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&0&1&1&1&1&0&1&1&1&1&1&1&0&1&1
\end{pmatrix}$$

Interconnections by PN places

$$\mathbf{F} = \begin{pmatrix} \text{blockdiag}(\mathbf{F}_i)_{i=1,N_A} \\ \text{----------} \\ \mathbf{F}_d \end{pmatrix}$$

$$\mathbf{G} = \begin{pmatrix} \text{blockdiag}(\mathbf{G}_i)_{i=1,N_A} & | \, \mathbf{G}_d \end{pmatrix}$$

$$\mathbf{B} = \begin{pmatrix} \text{blockdiag}(\mathbf{B}_i)_{i=1,N_A} \\ \text{----------} \\ \mathbf{B}_d \end{pmatrix}$$

where $\mathbf{B}_i = \mathbf{G}_i^T - \mathbf{F}_i$; $\mathbf{B}_{d_i} = \mathbf{G}_{d_i}^T - \mathbf{F}_{d_i}$; $i = 1, \ldots, N_A$;

$\mathbf{F}_d = (\mathbf{F}_{d_1}, \mathbf{F}_{d_2}, \ldots, \mathbf{F}_{d_{N_A}})$; $\mathbf{G}_d = (\mathbf{G}_{d_1}^T, \mathbf{G}_{d_2}^T, \ldots, \mathbf{G}_{d_{N_A}}^T)^T$

$\mathbf{B}_d = (\mathbf{B}_{d_1}, \mathbf{B}_{d_2}, \ldots, \mathbf{B}_{d_{N_A}})$.

Here $\mathbf{F}_i$, $\mathbf{G}_i$, $\mathbf{B}_i$ represent the parameters of the PN-based model of A_i.

$\mathbf{F}_d$, $\mathbf{G}_d$, $\mathbf{B}_d$ represent the structure of the interface between the agents.

This interface consists (exclusively) of additional PN places.

Interconnection by PN transitions and PN places

The interface is the PN subnet (quasi another agent) with n_d places, m_c transitions.

$$\mathbf{F} = \left(\begin{array}{c|c} \text{blockdiag}(\mathbf{F}_i)_{i=1,N_A} & \mathbf{F}_c \\ \hline \mathbf{F}_d & \mathbf{F}_{d\leftrightarrow c} \end{array} \right)$$

$$\mathbf{G} = \left(\begin{array}{c|c} \text{blockdiag}(\mathbf{G}_i)_{i=1,N_A} & \mathbf{G}_d \\ \hline \mathbf{G}_c & \mathbf{G}_{c\leftrightarrow d} \end{array} \right)$$

$$\mathbf{B} = \left(\begin{array}{c|c} \text{blockdiag}(\mathbf{B}_i)_{i=1,N_A} & \mathbf{B}_c \\ \hline \mathbf{B}_d & \mathbf{B}_{d\leftrightarrow c} \end{array} \right)$$

where $\mathbf{B}_i = \mathbf{G}_i^T - \mathbf{F}_i$; $\mathbf{B}_{d_i} = \mathbf{G}_{d_i}^T - \mathbf{F}_{d_i}$; $\mathbf{B}_{c_i} = \mathbf{G}_{c_i}^T - \mathbf{F}_{c_i}$; $i = 1, \ldots, N_A$; $\mathbf{B}_{d\leftrightarrow c} = \mathbf{G}_{c\leftrightarrow d}^T - \mathbf{F}_{d\leftrightarrow c}$.

$\mathbf{F}_{d\leftrightarrow c}$, $\mathbf{G}_{c\leftrightarrow d}$, $\mathbf{B}_{d\leftrightarrow c}$ are the structural matrices of the interface kernel.

Supervision (supervisory control) provides a theoretical framework for the automatic control of DES.

The theory of supervisory control of DES was introduced for designing controllers so that the controlled system satisfies certain desired qualitative constraints - e.g. a buffer in a manufacturing system should never overflow, or a message sequence in a communication network must be received in the same order as it was transmitted, etc.

Here, two kinds of PN-based supervision will be presented. Namely,

- supervision based on PN place invariants (P-invariants)
- generalized supervision utilizing PN places, transitions and the Parikh's vector

4. Agent Cooperation based on Modularity & Supervision

A supervisor is used here to avoid the egoistic effort of autonomous agents (when limited sources - e.g. working space, raw materials or semiproducts, energy, etc.).

By means of prohibition some states of the global system a useless 'haggle' of agents each other for a priority can be removed on behalf of the global system purposes.

On the contrary, the supervision process can be understood to be a carrier (performer) of the cooperation wrt. the global system politics.

Thus, the conditions for the supervisor synthesis represent the desired cooperation of agents in a group of agents or in MAS.

Some constraints has to be satisfied in order to achieve the desired behaviour (i.e. to synthesize the supervisor).

Two kinds of constrains known from supervising methodology in DES control theory will be considered:

(i) the constraints based on the P-invariants

(ii) the generalized constraints based also on the PN Parikh's vector and/or on the PN transitions.

The principle of the method is based on the PN P-invariants.

P-invariants are the vectors, $\mathbf{v}$, with the property that multiplication of these vectors with any state vector $\mathbf{x}_k \in X_{reach}$ (i.e. reachable from a given initial state vector $\mathbf{x}_0 \in X_{reach}$) yields the same result.

It is the relation of the state conservation:

$$\begin{aligned} \mathbf{v}^T.\mathbf{x}_k &= \mathbf{v}^T.\mathbf{x}_0 \\ \mathbf{v}^T.\mathbf{x}_k &= \mathbf{v}^T.\mathbf{x}_0 + \mathbf{v}^T.\mathbf{B}.\mathbf{u}_{k-1} \end{aligned}$$

Hence, to satisfy the previous definition of P-invariants, the condition

$$\mathbf{v}^T.\mathbf{B} = \mathbf{0}$$

has to be met.

P-invariants are useful in checking the property of mutual exclusion.

To eliminate a selfish behaviour of autonomous agents at exploitation of limited joint resources it is necessary to allocate the sources to individual agents rightly, with respect to the global goal of MAS.

Such a constraint of the agents behaviour and violation of their autonomy is rather in favour of MAS than in disfavour.

In case of the existence of several (e.g. n_x) invariants in a PN, the set of the P-invariants is created by the columns of the $(n \times n_x)$-dimensional matrix $\mathbf{V}$ being the solution of the homogeneous system of equations

$$\mathbf{V}^T.\mathbf{B} = \emptyset$$

This equation represents the base for the supervisor synthesis method.

Some additional PN places (slacks) can be added to the PN-model in question. The slacks create the places of the supervisor. Hence, the previous equation can be rewritten into the form

$$[\mathbf{L} \;\; \mathbf{I}_s].\begin{bmatrix} \mathbf{B} \\ \mathbf{B}_s \end{bmatrix} = \emptyset$$

where $\mathbf{I}_s$ is $(n_s \times n_s)$-dimensional identity matrix with $n_s \leq n_x$ being the number of slacks, $(n_s \times n)$-dimensional matrix $\mathbf{L}$ of integers represents (in a suitable form) the conditions

$$\mathbf{L}.\mathbf{x} \leq \mathbf{b} \quad \Rightarrow \; [\mathbf{L} \;\; \mathbf{I}_s].\begin{bmatrix} \mathbf{x} \\ \mathbf{x}_s \end{bmatrix} = \mathbf{b}$$

imposed on marking of the original PN (where $\mathbf{b}$ is the vector of integers), and $\mathbf{B}_s$ is $(n_s \times m)$-dimensional matrix representing (after its finding by computing) the structure of the PN-based model of the supervisor. Hence,

$$\mathbf{L}.\mathbf{B} + \mathbf{B}_s = \emptyset; \quad \mathbf{B}_s = -\mathbf{L}.\mathbf{B}; \quad \mathbf{B}_s = \mathbf{G}_s^T - \mathbf{F}_s$$

The augmented state vector (i.e. the state vector of the original PN together with the supervisor) and the augmented matrices are as follows

$$\mathbf{x}_a = \begin{bmatrix} \mathbf{x} \\ \mathbf{x}_s \end{bmatrix}; \;\; \mathbf{F}_a = \begin{pmatrix} \mathbf{F} \\ \mathbf{F}_s \end{pmatrix}; \;\; \mathbf{G}_a^T = \begin{pmatrix} \mathbf{G}^T \\ \mathbf{G}_s^T \end{pmatrix}$$

where the submatrices $\mathbf{F}_s$ and $\mathbf{G}_s^T$ correspond to the interconnections of the incorporated slacks with the actual PN structure. Because of the prescribed conditions we have

$$[\mathbf{L} \mid \mathbf{I}_s].\begin{bmatrix} \mathbf{x}_0 \\ {}^s\mathbf{x}_0 \end{bmatrix} = \mathbf{b} \quad \text{i.e. the supervisor initial state is: } {}^s\mathbf{x}_0 = \mathbf{b} - \mathbf{L}.\mathbf{x}_0$$

where $\mathbf{b}$ is the vector of the corresponding dimensionality (i.e. n_s) with integer entries representing the limits for number of common tokens - i.e. the maximum numbers of tokens that the corresponding places can possess altogether (i.e. share).

Example 4.1

Let us show how easy the Dijkstra's 'dining philosophers' problem can be solved by means of the supervisor synthesis. It is a classic multi-process synchronization problem where five computers competed for access to five shared tape drive peripherals.

Namely, five philosophers are sitting at a circular table with a large bowl of spaghetti in the center doing one of two activities - eating or thinking. While eating, they are not thinking, and while thinking, they are not eating. A chopstick is placed in between each philosopher.

Each philosopher has one chopstick to his left and one chopstick to his right. It is assumed that a philosopher must eat with two chopsticks. The philosopher can only use the chopstick on his immediate left or right.

The PN-based model of the situation for one philosopher is given as

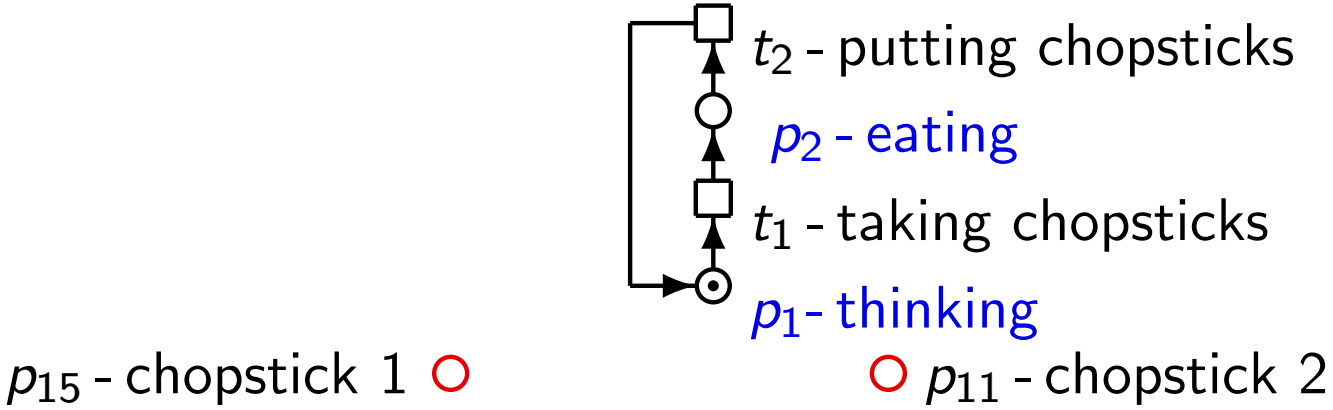

Figure 10. *The PN-based model of one philosopher activities.*

In case of five philosophers the thinking is modelled by the PN places p_1, p_3, p_5, p_7, p_9 and eating is represented by the places p_2, p_4, p_6, p_8, p_{10}. In this situation all of the philosophers are thinking - p_1, p_3, p_5, p_7, p_9 are active - i.e. no forks are necessary. However, formally they are expressed by means of the PN places p_{11}, p_{12}, p_{13}, p_{14}, p_{15}, apart from interconnections.

The defined problem can be solved by the supervisor synthesis method. The incidence matrices of the PN models of the autonomous agents A_i, $i = 1, ..., 5$ are

$$\mathbf{F}_i = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix};\ \mathbf{G}_i^T = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix};\ \mathbf{B}_i = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$$

Consider that the initial states are the same

$${}^i\mathbf{x}_0 = (1,\, 0)^T;\ \ i = 1, ..., 5$$

The parameters of the PN model of the group of autonomous agents can be expressed as follows

$$\mathbf{F} = \text{blockdiag}(\mathbf{F}_i)_{i=1,5};\ \mathbf{G} = \text{blockdiag}(\mathbf{G}_i)_{i=1,5}$$

$$\mathbf{x}_0 = ({}^1\mathbf{x}_0^T, {}^2\mathbf{x}_0^T, {}^3\mathbf{x}_0^T, {}^4\mathbf{x}_0^T, {}^5\mathbf{x}_0^T)^T$$

The conditions imposed on the autonomous agents are

$$\begin{aligned}
\sigma_{p_2} + \sigma_{p_4} &\leq 1 \\
\sigma_{p_4} + \sigma_{p_6} &\leq 1 \\
\sigma_{p_6} + \sigma_{p_8} &\leq 1 \\
\sigma_{p_8} + \sigma_{p_{10}} &\leq 1 \\
\sigma_{p_{10}} + \sigma_{p_2} &\leq 1
\end{aligned}$$

Verbally it means that two adjacent agents (neighbours) must not eat simultaneously. These conditions yield the matrix **L** and the vector **b** as follows

$$\mathbf{L} = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}; \quad \mathbf{b} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Hence,

$$\mathbf{B}_s = -\mathbf{L}.\mathbf{B}; \; {}^s\mathbf{x}_0 = \mathbf{b} - \mathbf{L}.\mathbf{x}_0$$

$$\mathbf{B}_s = \begin{pmatrix} -1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & 1 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix}; \; {}^s\mathbf{x}_0 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\mathbf{F}_s = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} ; \mathbf{G}_s^T = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

The structural matrices $\mathbf{F}_s$, $\mathbf{G}_s$ of the supervisor give us the structural interconnections between the philosophers and the forks. Using the supervisor synthesis the problem was easily resolved. The PN-based model of the solution - the cooperating agents - is given in **Figure 2**.

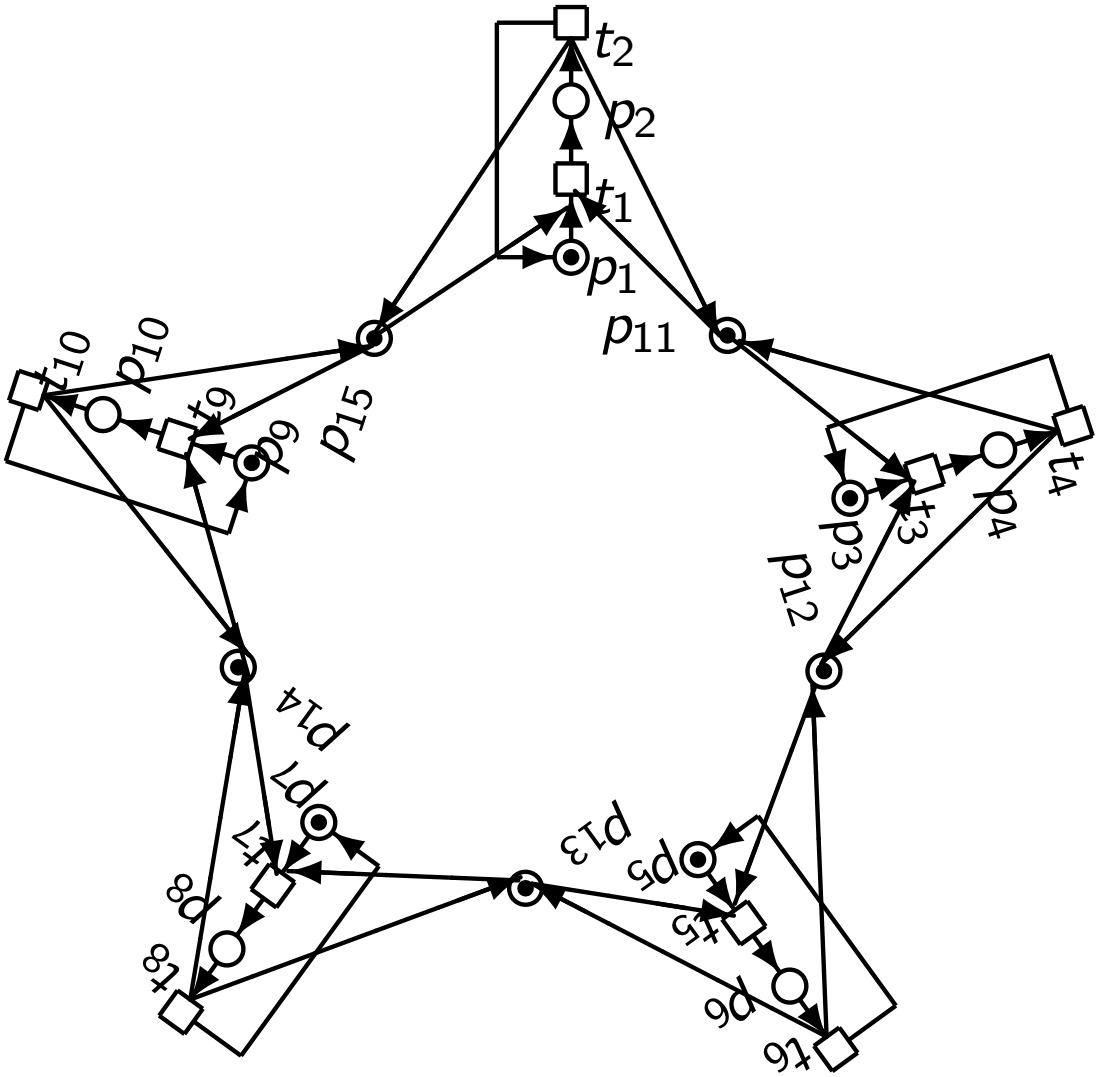

Figure 11. *The PN-based model of the supervised dining philosophers.*

Example 4.2

The conditions for cooperation can be more complicated. Consider e.g. the group of 5 simple autonomous agents $Gr_A = \{A_1, A_2, A_3, A_4, A_5\}$ with the same structure like those handled above.

Solve the situation when it is necessary to ensure that only one agent from each of the subgroups $Sgr_1 = \{A_1, A_4, A_5\}$, $Sgr_2 = \{A_2, A_4, A_5\}$, and $Sgr_3 = \{A_3, A_4, A_5\}$ can simultaneously cooperate with other agents from Gr_A. In other words, the agents inside the designated subgroups must not work simultaneously.

Even, the agents A_4 and A_5 can work only individually (any cooperation with other agents is excluded). However, the agents A_1, A_2, A_3 can work simultaneously.

Now, the conditions prescribing the cooperation of agents are

$$\sigma_{p_2} + \sigma_{p_8} + \sigma_{p_{10}} \leq 1$$
$$\sigma_{p_4} + \sigma_{p_8} + \sigma_{p_{10}} \leq 1$$
$$\sigma_{p_6} + \sigma_{p_8} + \sigma_{p_{10}} \leq 1$$

It means

$$\mathbf{L} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}; \quad \mathbf{b} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

After the supervisor synthesis the PN model of the cooperating agents is displayed in **Figure 3**.

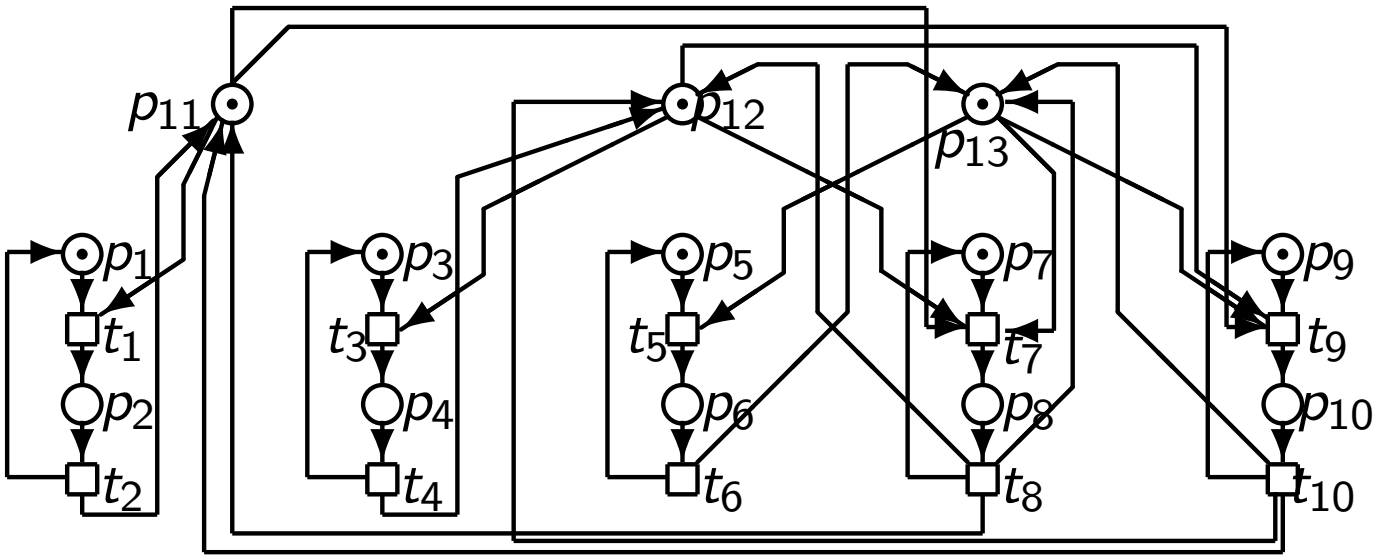

Figure 12. *The PN-based model of the 3 groups cooperation*

To widen a class of cooperation the more general approach can be used. On this way also the Parikh's vector is very important and useful. The general linear constraints for supervisor synthesis are

$$\mathbf{L}_p.\mathbf{x} + \mathbf{L}_t.\mathbf{u} + \mathbf{L}_v.\mathbf{v} \leq \mathbf{b}$$

where $\mathbf{L}_p$, $\mathbf{L}_t$, $\mathbf{L}_v$ are, respectively, $(n_s \times n)-$, $(n_s \times m)-$, $(n_s \times m)-$dimensional matrices. When $\mathbf{b} - \mathbf{L}_p.\mathbf{x} \geq \mathbf{0}$ is valid the supervisor with the following structure and initial state

$$\begin{aligned} \mathbf{F}_s &= \max(\mathbf{0},\ \mathbf{L}_p.\mathbf{B} + \mathbf{L}_v,\ \mathbf{L}_t);\ \mathbf{L}_{pv} = \mathbf{L}_p.\mathbf{B} + \mathbf{L}_v \\ \mathbf{G}_s^T &= \max(\mathbf{0},\ \mathbf{L}_t - \max(\mathbf{0},\ \mathbf{L}_{pv})) - \min(\mathbf{0},\ \mathbf{L}_{pv}) \\ {}^s\mathbf{x}_0 &= \mathbf{b} - \mathbf{L}_p.\mathbf{x}_0 - \mathbf{L}_v.\mathbf{v}_0 \end{aligned}$$

guarantees that constraints are verified for the states resulting from the initial state. Here, the max(.) is the maximum operator for matrices. However, the maximum is taken element by element.

Developing the model of PN dynamics we have

$$\begin{aligned}
\mathbf{x}_1 &= \mathbf{x}_0 + \mathbf{B}.\mathbf{u}_0 \\
\mathbf{x}_2 &= \mathbf{x}_1 + \mathbf{B}.\mathbf{u}_1 = \mathbf{x}_0 + \mathbf{B}.(\mathbf{u}_0 + \mathbf{u}_1) \\
& \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
\mathbf{x}_k &= \mathbf{x}_0 + \mathbf{B}.\sum_{i=0}^{k-1} \mathbf{u}_i = \mathbf{x}_0 + \mathbf{B}.\mathbf{v}
\end{aligned}$$

where just the vector $\mathbf{v} = \sum_{i=0}^{k-1} \mathbf{u}_i$ of integers is named to be the Parikh's vector.

This vector gives us information about how many times the particular transitions are fired during the development of the system dynamics from the state $\mathbf{x}_0$ to the state $\mathbf{x}_k$.

The Parikh's vector can also be utilized in the DES supervisor synthesis. Namely, together with the state vector $\mathbf{x}$ and the control vector $\mathbf{u}$.

Case Study

Let us illustrate the approach on the case of the internal transport of a flexible manufacturing system (FMS).

Combining both kinds of constraints will be used step-by-step in order to synthesize the supervisor.

The agents working in a common space - the tracks for AGVs (automatically guided vehicles) in a kind of FMS - have to be supervised in order to avoid a crash.

To illustrate this, consider N_t tracks of AGVs in FMS. Denote them as agents A_i, $i = 1, ..., N_t$.

The AGVs carry semi-products from a place of FMS to another place and then they (empty or with another load) come round.

In any track A_i there exist $n_i \geq 1$ AGVs.

The PN model of the single agent A_1 is given in **Figure 4**.

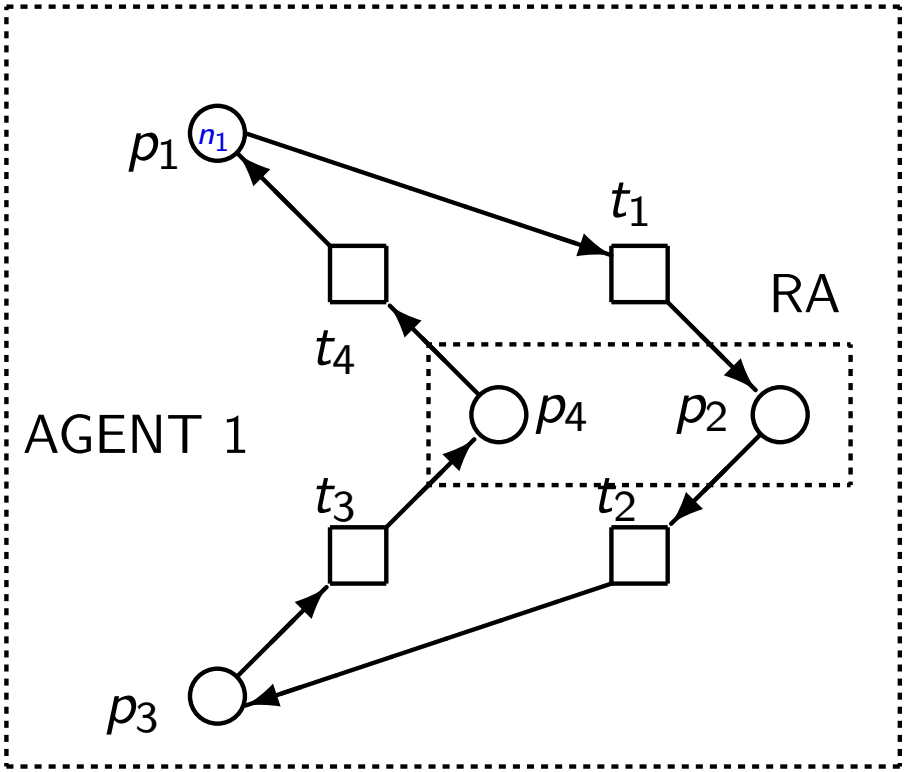

Figure 13. *The PN-based model of the agent. The places p_2, p_4 lie in the RA*

The parameters of the agents PN-based models are the following

$$\mathbf{F}_i = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}; \mathbf{G}_i^T = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}; \ i = 1, \ N_t$$

During the agents activities n_1 AGVs have to pass this track as well as a restricted area (RA) common for all agents, namely, even two times.

RA is a "bottle-neck" of the global system. Namely, in case of the AGVs of e.g. the agent A_1:

(i) when they carry some semi-products from a place p_1 of FMS to another place p_3 they have to pass RA (expressed by p_2) first time

(ii) when they come round to the place p_1 they have to pass RA (expressed now by p_4) once more.

However, because the space of the FMS where the agents operate is limited, there exists the restriction that only limited number of different AGVs, namely

$$N < \sum_{i=1}^{N_t} n_i \quad \text{or often} \quad N << \sum_{i=1}^{N_t} n_i$$

can operate in the RA simultaneously, the agents A_i have to be limited in their autonomous activities by a supervisor.

The reason is that the agents themselves are not able to coalesce on a procedure satisfying all of them because the autonomous agents are usually egoistic (selfish).

A violent driving of individual agents in RA might tend to wrecks with exterminatory effects, including some mechanical devastations, even standing the whole FMS off.

Therefore, the supervisor determines a policy of the agents behaviour from the global point of view in order to achieve the satisfying results of the cooperative interactions among devices and expected behaviour (function) of the global FMS.

The opposite view on the supervisor synthesis process can evoke an impress that such a process expresses e.g. the agents negotiation (although unwilling) or another kind of cooperation.

The supervisor does not drive its own selfish will or interest but its activity represents only the necessary part of the global strategy of the FMS behaviour, even the correct model of a part of the technological subprocess inside FMS.

From the control theory point of view the supervisor realizes the objective function of the FMS subprocess.

Namely, the supervisor only realizes the global demands on the behaviour of a part of FMS so as to meet the global aim of the whole FMS.

Considering N_A agents, the restrictions in analytical terms are

$$\begin{aligned}
&\sigma_{p_2} + \sigma_{p_4} && \leq n1 \\
&\qquad\qquad \sigma_{p_6} + \sigma_{p_8} && \leq n2 \\
&\ldots \qquad \ldots \qquad \ldots \qquad \ldots && \\
&\qquad\qquad\qquad\qquad \sigma_{p_{4N_A-2}} + \sigma_{p_{4N_A}} && \leq n_{N_A} \\
&\sigma_{p_2} + \sigma_{p_4} + \sigma_{p_6} + \sigma_{p_8} + \ldots + \sigma_{p_{4N_A-2}} + \sigma_{p_{4N_A}} && \leq N
\end{aligned}$$

When $N_A = 4$, $N = 2$, $n_1 = n_2 = n_3 = n_4 = 1$ then

$$\begin{aligned}
&\sigma_{p_2} + \sigma_{p_4} && \leq 1 \\
&\qquad\qquad \sigma_{p_6} + \sigma_{p_8} && \leq 1 \\
&\qquad\qquad\qquad\qquad \sigma_{p_{10}} + \sigma_{p_{12}} && \leq 1 \\
&\qquad\qquad\qquad\qquad\qquad\qquad \sigma_{p_{14}} + \sigma_{p_{16}} && \leq 1 \\
&\sigma_{p_2} + \sigma_{p_4} + \sigma_{p_6} + \sigma_{p_8} + \sigma_{p_{10}} + \sigma_{p_{12}} + \sigma_{p_{14}} + \sigma_{p_{16}} && \leq 2
\end{aligned}$$

$$\mathbf{L} = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}; \quad \mathbf{b} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 2 \end{pmatrix}$$

$$\mathbf{F}_s = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

$$\mathbf{G}_s^T = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

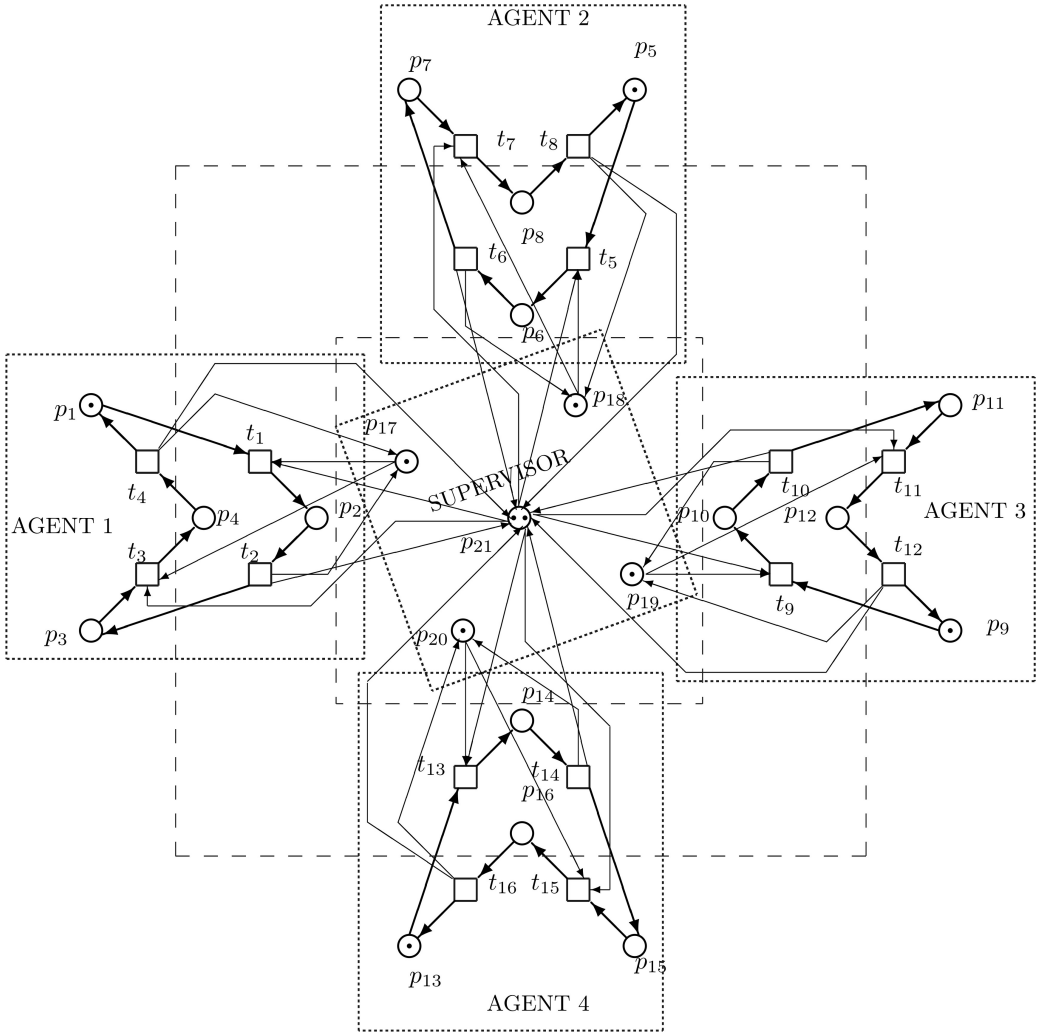

Figure 14. *The PN-based model of supervising 4 agents in order to simultaneously exploit the RA.*

In such a supervisor structure only the presence of N AGVs in the RA simultaneously is assured without designation which agents ($N = 2$ agents from 4 existing ones) have the priority to enter by their AGV into the area.

To resolve this problem it is necessary to ensure priorities.

Especially, in the given initial state when all of the agents compete for entering the area, it is necessary to choose N of the N_t agents.

During the global FMS dynamics development it is probable that not all of the agents will compete for entering. But more than N agents can compete.

However, there is impossible in a real FMS to presume that the agents will negotiate each other to find the global optimum.

Usually, there is no time for such a "democratic" negotiation process.

However, we can synthesize another supervisor for the system being already supervised by the existing supervisor synthesized above.

The advantage of such a multilevel approach consists in a flexibility.

While the 1th level supervisor assures the stable situation that only two AGVs can occur in the RA, the 2nd level supervisor can determine on which track (i.e. to which agent A_i AGVs belong in).

In general, when we want to enter priorities, the new supervisor can be synthesized.

We can consider e.g. that the priorities π_{A_i} of agents A_i descends with the ascending agent number - i.e. $\pi_{A_1} > \pi_{A_2} > \pi_{A_3} > \pi_{A_4}$.

The Agent 1 has the highest priority as to entering to RA.

The priorities of other agents descend with ascending number denoting the agent in question, namely in both directions.

The constraints imposed on elements of the Parikh's vector are

$$v_5 \leq v_1;\ v_9 \leq v_1;\ v_{13} \leq v_1;\ v_6 \leq v_1;\ v_{10} \leq v_1;\ v_{14} \leq v_1$$
$$v_9 \leq v_5;\ v_{13} \leq v_5;\ v_{10} \leq v_5;\ v_{14} \leq v_5;\ v_{13} \leq v_9;\ v_{14} \leq v_9.$$

Considering $\mathbf{v}_0 = \mathbf{0}$, $\mathbf{b} = \mathbf{0}$ and respecting the constraints expressed by

$$\mathbf{L}_v = \begin{pmatrix} -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

the following structure of the supervisor is obtained

$$
{}^{(2)}\mathbf{F}_s = \begin{pmatrix}
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0
\end{pmatrix}; \quad {}^{(2)s}\mathbf{x}_0 = \begin{pmatrix}
0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0
\end{pmatrix}
$$

$$
{}^{(2)}\mathbf{G}_s^T = \begin{pmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0
\end{pmatrix}
$$

Hence, the initial state of the resulting supervised system is

$$^{(2)}\mathbf{x}_0 = (\mathbf{x}_{a0}^T, \ ^{(2)s}\mathbf{x}_0^T)^T$$

where $\mathbf{x}_{a0} = (1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0|1,1,1,1,2)^T$.

Respecting the structure of the augmented system supervised by the 1st supervisor, the structure of the fully supervised system (i.e. by both supervisors) is the following

$$^{(2)}\mathbf{F}_a^T = \left(\ (\mathbf{F}^T|\mathbf{F}_s^T)^T|^{(2)}\mathbf{F}_s^T \ \right)^T ; \ ^{(2)}\mathbf{G}_a = \left(\ (\mathbf{G}|\mathbf{G}_s)|^{(2)}\mathbf{G}_s \ \right)$$

Here, $^{(2)}(.)$ expresses that the matrices/vectors belonging to the 2nd supervisor are meant. The 2nd supervisor is synthesized for the augmented system (i.e. the original agents already supervised by the first supervisor).

The structure of the 2nd supervisor is given in **Figure 5**.

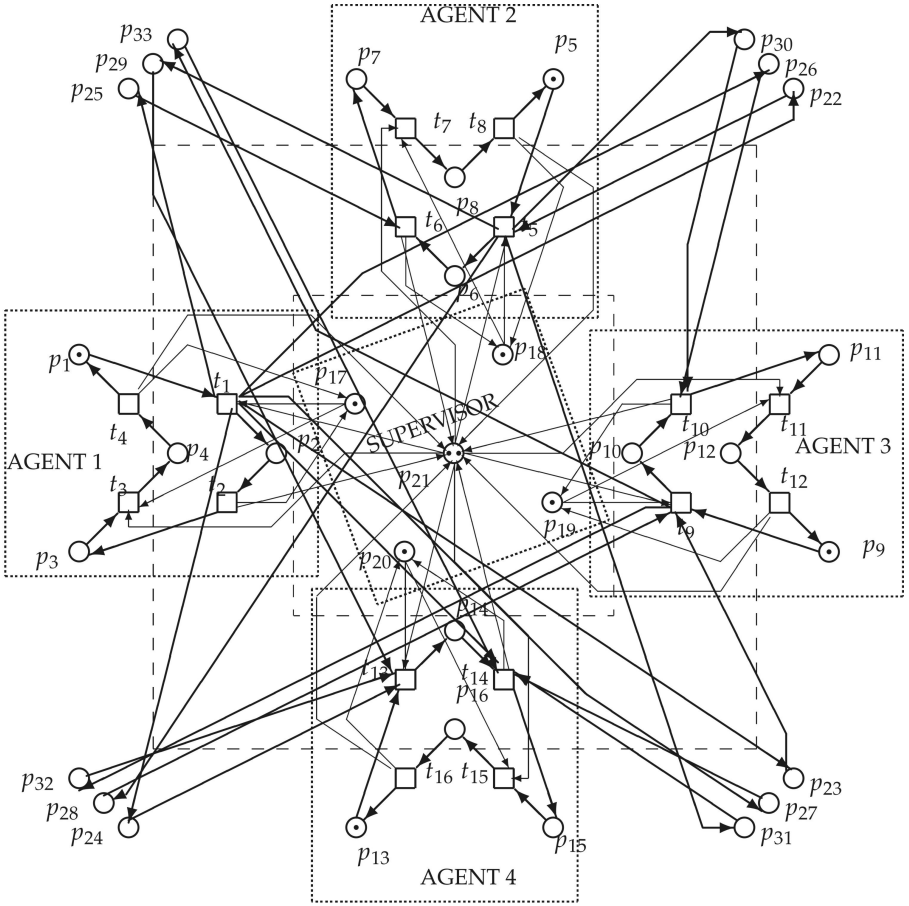

Figure 15. *The PN-based model of supervising after embedding the 2nd supervisor.*

5. Hybrid Agents and their Cooperation

Continuous systems (CS) are usually described by means of differential equations describing processes at respecting physical laws.

However, there are certain complex CS where it is practically impossible or very difficult to obtain a CS model corresponding to the real system.

Two main difficulties occur on that way:
(i) how to determine the kind of differential equations describing the particular CS - namely, to guess an order of the system when the linear differential equations are used or a kind of nonlinear differential equations

(ii) how to identify all parameters of the chosen kind of differential equations describing the real complex system by means of measuring (if any such parameters are measurable).

On that account other methods are found.

... Hybrid Agents and their Cooperation

Hybrid Petri nets (HPN) are frequently used for modelling complex hybrid systems (HS). Especially, the First Order Hybrid Petri Nets (FOHPN) seem to be very suitable for HS modelling.

HPN in general combine continuous Petri nets (CPS) with different kinds of Petri nets (PN) - like place/transition Petri nets (P/T PN), deterministic timed PN, stochastic PN, etc.

Because any CS has minimally two discrete states - it is either working or idle - the mutual transitions between these states are discrete events .

Thus, the idea arose:

(i) to create the model of hybrid autonomous agents (e.g. production lines in complex manufacturing systems) by means of FOHPN

(ii) then to synthesize the cooperation of the agents by means of discrete-event systems (DES) supervision.

HPN are able to model the coexistence of discrete and continuous variables. This brings following advantages:

(i) reducing the dimensionality of the state space
(ii) increasing the computational efficiency of the simulation process
(iii) defining optimization problems of polynomial complexity.

The places, transitions and oriented arcs consist of two groups - discrete and continuous.
Moreover, beside the arcs among discrete places and discrete transitions and the arcs among continuous places and continuous transitions there exist the arcs among discrete places and continuous transitions as well as the arcs among the continuous places and discrete transitions.

... Hybrid Agents and their Cooperation

... First Order Hybrid Petri Nets (FOHPN)

The discrete places and transitions handle discrete tokens while the continuous places and transitions handle continuous variables (e.g. like different kinds of material flows).

The mutual interaction between these groups is possible according to prescribed rules.

The set $\mathcal{P}$ of places consists of two subsets $\mathcal{P} = \mathcal{P}_d \cup \mathcal{P}_c$, where

$\mathcal{P}_d$ is a set of discrete places (graphically represented by simple circles) and

$\mathcal{P}_c$ is a set of continuous places (represented usually by double concentric circles).

Cardinalities of the sets are, respectively, n, n_d and n_c.

... Hybrid Agents and their Cooperation

... First Order Hybrid Petri Nets (FOHPN)

The set of transitions $\mathcal{T}$ consists of two subsets $\mathcal{T} = \mathcal{T}_d \cup \mathcal{T}_c$, where

$\mathcal{T}_d$ is a set of discrete transitions (graphically represented by simple rectangles) and

$\mathcal{T}_c$ is a set of continuous transitions (represented usually by double rectangles - a smaller rectangle inside of the bigger one).

Their cardinalities are, respectively, q, q_d and q_c.

Moreover, $\mathcal{T}_d$ can contain a subset of immediate transitions (like in ordinary PN) and/or a subset of timed transitions. The timed transitions express the behaviour of discrete events in time and they may be deterministic and/or stochastic.

To ensure qualitative properties of FOHPN (so called *well-formed nets*): Firing of continuous transitions must not influence marking of discrete places.

FOHPN marking is a function assigning a non-negative integer number of tokens to each of the discrete places and an amount of fluid to each of the continuous places.

To each of the continuous transition t_j an instantaneous firing speed (IFS) is assigned. IFS determines an amount of fluid per a time unit (i.e. a sort of the flow rate) which fires the continuous transition in a time instance τ.

For all of the time instances τ holds $V_j^{min} \leq v_j(\tau) \leq V_j^{max}$, where *min* and *max* denote the minimal and maximal values of the speed $v_j(\tau)$.

Consequently, IFS of any continuous transition is piecewise constant.
An empty continuous place p_i is filled through its enabled input transition.
In such a way the fluid can flow to the output transition of this place.

The continuous transition t_j is enabled in the time τ iff

(i) its input discrete places $p_k \in \mathcal{P}_d$ have marking $m_k(\tau)$ at least equal to $Pre(p_k, t_j)$ (**Pre** and **Post** are the incidence matrices well known in PN - above denoted as $\mathbf{F}$ and $\mathbf{G}^T$)

(ii) and all of its input continuous places $p_i \in \mathcal{P}_c$ satisfies the condition that either $m_i(\tau) \geq 0$ or the place p_i is filled.

If all of the input continuous places of the transition t_j have non-zero marking then t_j is strongly enabled, otherwise t_j is weakly enabled.

The continuous transition t_j is disabled if some of its input places is not filled.

Marking development of the continuous place

In general, the marking development of the continuous place $p_i \in \mathcal{P}_c$ in time can be described by the differential equation

$$\frac{dm_i}{d\tau} = \sum_{t_j \in T_c} C(p_i,\ t_j).v_j(\tau) \tag{1}$$

where

$v_j(\tau)$ are entries of the IFS vector $\mathbf{v}(\tau) = (v_j(\tau), \cdots, v_{n_c}(\tau))^T$ in the time τ

$\mathbf{C}$ is the incidence matrix of the continuous part of FOHPN (i.e. the matrix $\mathbf{C} = \mathbf{Post} - \mathbf{Pre}$).

The differential equation holds provided that no discrete transition is fired in the time τ and all of the speeds $v_j(\tau)$ are continuous in the time τ.

The IFS $v_j(\tau)$, $j = 1, \ldots, n_c$, defines enabling the continuous transition t_j.

If t_j is strongly enabled then it can be fired with an arbitrary firing speed $v_j(\tau) \in [V_j^{min}, V_j^{max}]$.

If t_j is weakly enabled then it can be fired with an arbitrary firing speed $v_j(\tau) \in [V_j^{min}, V_j]$, where $V_j \leq V_j^{max}$.

Namely, t_j cannot take more fluid from any empty input continuous place than the amount entering the place from other transitions.
It corresponds to the principle of conservation of matter.

5.2 FOHPN Based Model of the Production Line

Consider the recycling line producing the plastic double foil from the granulate prepared from the waste plastic. The plastic foil is used for producing plastic bags. The rough FOHPN model is displayed in Fig. 16. To distinguish continuous and discrete places as well as the continuous and discrete transitions, the continuous items are denoted by capitals.

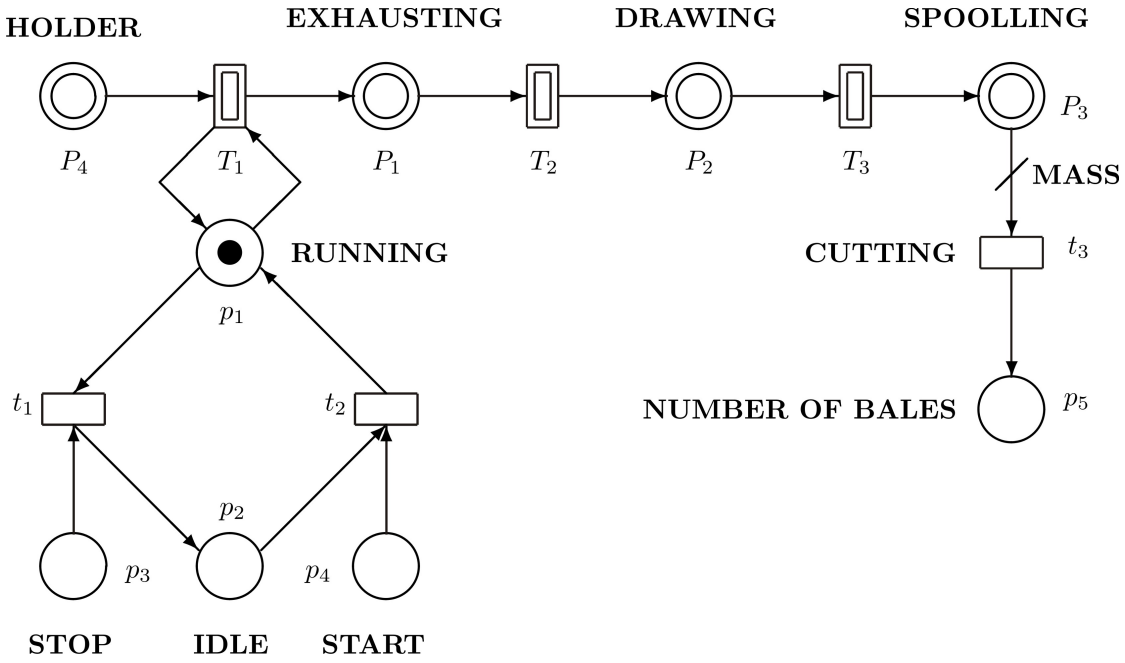

Figure 16. *The rough FOHPN-based model of the production line*

The granulate is collocated in the Holder represented by the place P_4.

Thence, the fluid flows through the transition T_1 to the Exhausting Machine (Exhauster) represented by the place P_1 where a big bubble is blow (in order to make producing the double foil possible).

Subsequently, the double foil is drawing into the prescribed width and thickness on the Drawing Line P_2

Drawn foil proceeds to the Spooling Machine P_3 where the bales of a prescribed Mass are prepared. Here, after achieving the prescribe Mass, the foil is aborted by cutting, the completed bale is withdrawn.

Then, the new bale starts to be spooled on a new spool. The completed bale proceeds to a buffer or directly to another production line where the bags are produced.

... Hybrid Agents and their Cooperation

... FOHPN Based Model of the Production Line

In the next line, the foil is enfolded, welds corresponding to the length of bags are performed, and the belt of bags is rolled into rolls with a uniform number of bags in each roll.

The marking of the discrete place p_5 expresses the number of bales produced by the line.

However, in the real machines time delays occur:
- because of the time necessary for producing the bubble in the Exhausting Machine as well as
- because of the transport delay of the Drawing Line

Therefore, it is necessary to build them into the FOHPN model.

Moreover, it is necessary to ensure regular supplying the granulate in order to avoid a breakdown of the line caused by the lack of the granulate.

Hence, the FOHPN model has the more detailed form given in Fig. 17 where the feedback $T_1 \rightarrow P_5 \rightarrow t_9 \rightarrow p_{15} \rightarrow t_8$ realizes supplying the granulate and

M_{fb} denotes the multiplicity of the arc due to added amount of the granulate in one batch.

N_g is marking of the discrete place p_{14} representing the number of the added batches of the granulate.

M_b represents the multiplicity of the arc corresponding to the prescribed mass of the bale and $\{p_6 - p_9,\ t_4,\ t_5\}$, $\{p_{10} - p_{13},\ t_6,\ t_7\}$ model, respectively, the delays of exhausting and drawing processes

$t_4 - t_7$ are timed transitions with delays

$p_8,\ p_9,\ p_{12},\ p_{13}$ fire these transitions.

The structure of the FOHPN is described by incidence matrices with indices cc, cd, dc, dd denoting incidences between places and transitions 'continuous-continuous', 'continuous-discrete', 'discrete-continuous', 'discrete-discrete'.

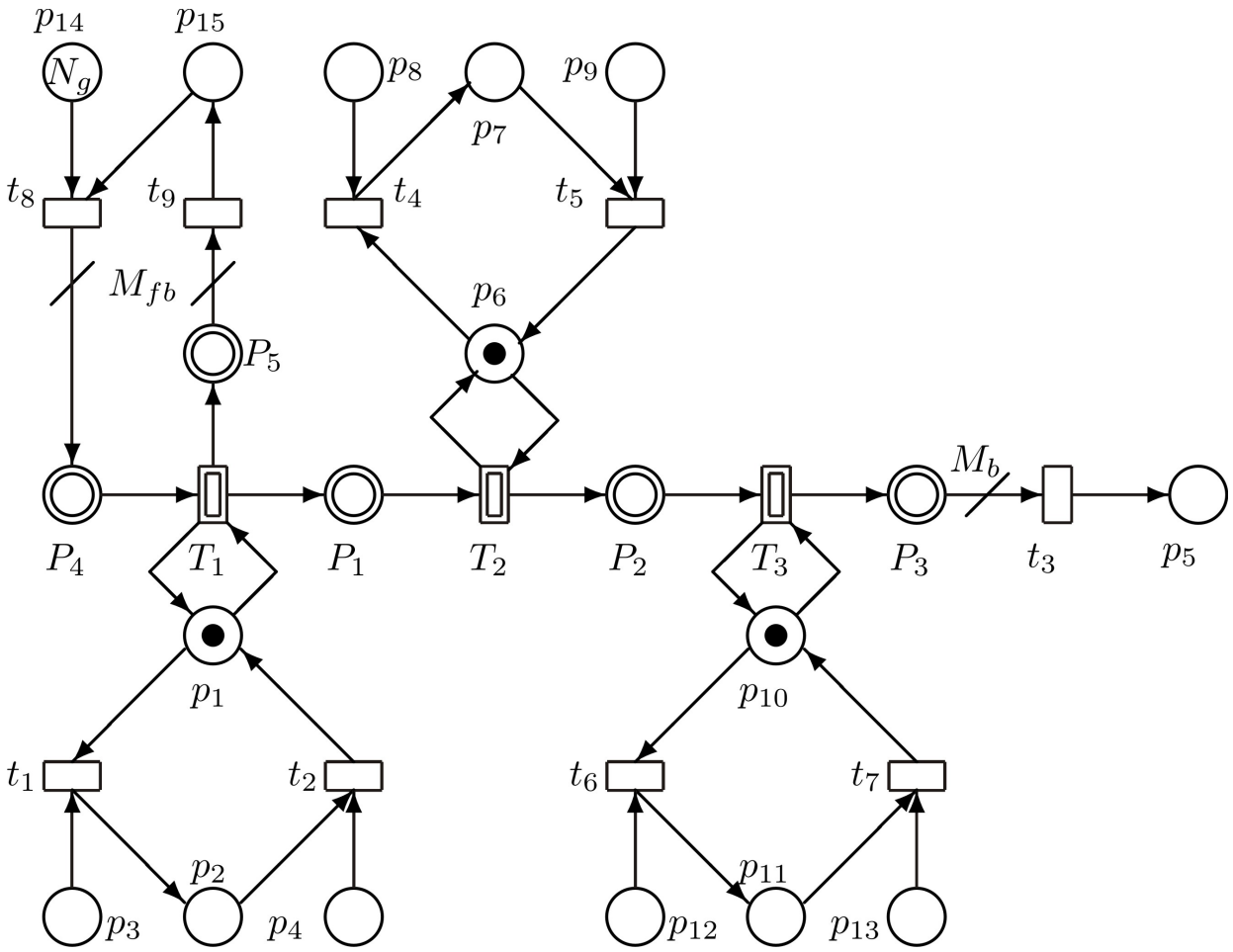

Figure 17. *The rough FOHPN-based model of the production line and a feedback corresponding to supplying the granulate*

$$\mathbf{Pre}_{cc} = \begin{pmatrix} 0\,1\,0 \\ 0\,0\,1 \\ 0\,0\,0 \\ 1\,0\,0 \\ 0\,0\,0 \end{pmatrix} ;\ \mathbf{Post}_{cc} = \begin{pmatrix} 1\,0\,0 \\ 0\,1\,0 \\ 0\,0\,1 \\ 0\,0\,0 \\ 1\,0\,0 \end{pmatrix}$$

$$\mathbf{Pre}_{cd} = \begin{pmatrix} 00\ 0\ 00000\ 0 \\ 00\ 0\ 00000\ 0 \\ 00\,M_b\,00000\ 0 \\ 00\ 0\ 00000\ 0 \\ 00\ 0\ 00000\,M_{fb} \end{pmatrix} ;\ \mathbf{Post}_{cd} = \begin{pmatrix} 0000000\ 0\ 0 \\ 0000000\ 0\ 0 \\ 0000000\ 0\ 0 \\ 0000000\,M_{fb}\,0 \\ 0000000\ 0\ 0 \end{pmatrix}$$

$$\mathbf{Pre}_{dc}^T = \begin{pmatrix} 1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0 \end{pmatrix} \equiv \mathbf{Post}_{dc}^T$$

$$\mathbf{Pre}_{dd} = \begin{pmatrix} 1&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0 \\ 0&0&0&1&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0 \\ 0&0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1&0 \end{pmatrix} ; \mathbf{Post}_{dd} = \begin{pmatrix} 0&1&0&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0 \\ 0&0&0&1&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1 \end{pmatrix}$$

The initial marking of the continuous places is

$$\mathbf{M}_c = (0,\ 0,\ 0,\ M_{gr},\ 0)^T$$

where M_{gr} is the initial amount of the granulate in P_4.

The initial marking of the discrete places is

$$\mathbf{m}_d = (1,\ 0,\ 1,\ 1,\ 0,\ 1,\ 0,\ 1,\ 1,\ 1,\ 0,\ 1,\ 1,\ N_g,\ 0)^T$$

where N_g is the number of the batches of the granulate to be added during the production.

Firing speeds of the continuous transitions $v_j(\tau)$, $j = 1, 2, 3$ are, respectively, from the intervals $[V_j^{min},\ V_j^{max}]$.

The discrete transitions are considered to be deterministic without any delay or with a transport delays mentioned above.

... Hybrid Agents and their Cooperation

... FOHPN Based Model of the Production Line

Using the Matlab simulation tool HYPENS (elaborated by the University of Cagliary, Italy) with

the structural parameters $M_b = 270,\ M_{fb} = 3750,$

the initial markings with $M_{gr} = 5000,\ N_g = 4,$

the limits of intervals for the firing speeds being
$V_j^{min} = 0,\ j = 1,\ 2,\ 3,\ V_1^{max} = 1.8,\ V_2^{max} = 1.5$ and $V_3^{max} = 1.4$

the delays of discrete transitions being the entries of the vector
$(0,\ 0,\ 0,\ 0.01,\ 125,\ 0.01,\ 300,\ 0,\ 0)^T$

we obtain the behaviour of the simulated line.

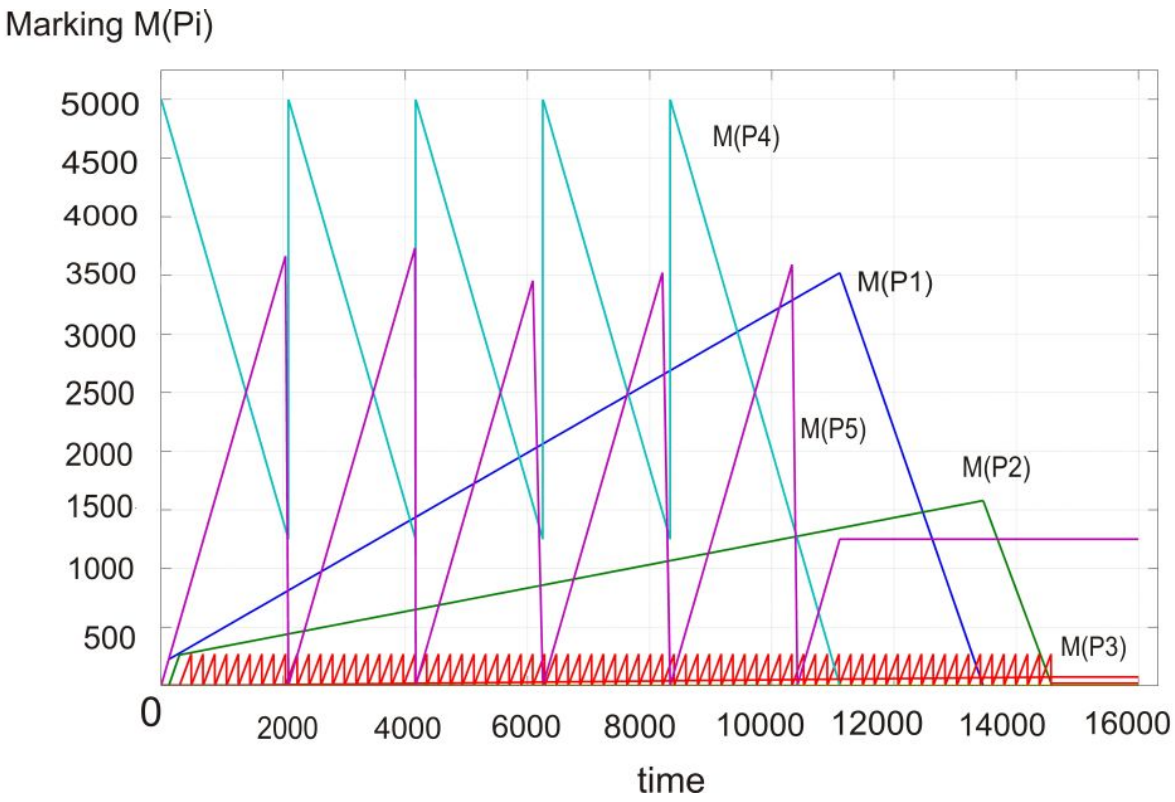

Figure 18.a. *The dynamics behaviour of the line material flows in the common scale.*

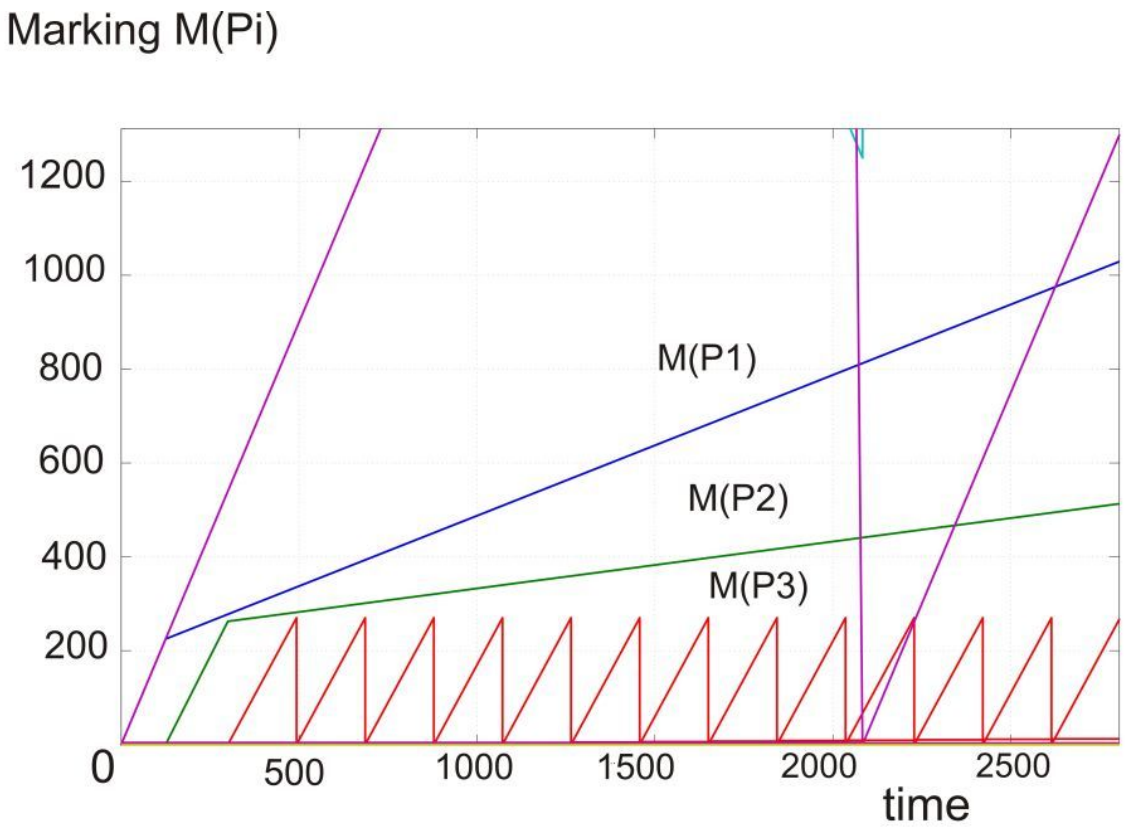

Figure 18.b. *The zoomed detail in order to see the transport delays better.*

In the Fig. 18.a the course of flows is shown during the time segment when the granules are added four times while the Fig 18.b displays the zoomed detail.

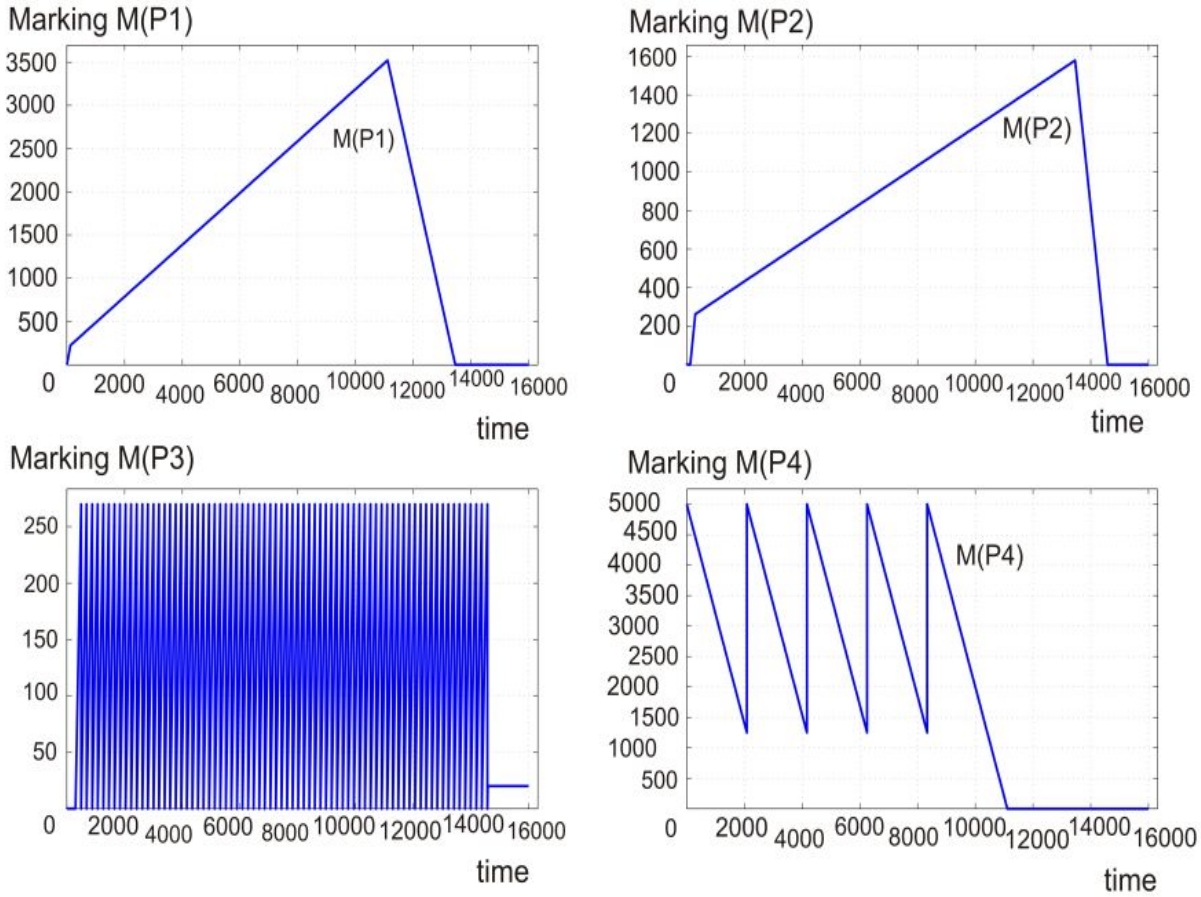

Figure 19. *The dynamics behaviour of the material flows of the production line in their individual scales - i.e. the marking evolution of the continuous places*

In Fig. 18.a, Fig. 18.b and Fig. 19 the results of the production line simulation are presented.
They illustrate the dynamic development of marking $M(P_i)$ of the continuous places P_i, $i = 1 \ldots, 5$, i.e. they draft the courses of the material flows throughout these places.

$M(P_5)$ models a feedback flow (a batch of granules is poured into the Holder in order to avoid its emptying),
$M(P_4)$ the flow of granules throughout the Holder,
$M(P_1)$ the flow of the exhausted bubble from the Exhauster,
$M(P_2)$ the flow of the drawing foil and
$M(P_3)$ spooling the particular bales of the foil. The finished bale is immediately removed in order to spool the next bale.

When the flow of granules is stopped, the other flows are gradually finished too.

... Hybrid Agents and their Cooperation

5.3 Cooperation of Hybrid Agents

Having a group of hybrid agents the cooperation can by synthesized by means of the DES supervision methods.

Consider N lines for the foil production and $M \leq N$ lines for producing the bags rolls. These lines are hybrid and they can be understood to be hybrid autonomous agents.

The lines producing the foil can work independently, even simultaneously in time. The same is valid for the lines producing the rolls of bags.

However, N lines producing the foil have to share only M rolling lines - $N < M$.

To solve this problem a strategy has to be defined for the sharing.

When such a strategy is defined the conditions for the supervisor synthesis

... Hybrid Agents and their Cooperation

... Cooperation of Hybrid Agents

The simplest form of the agents cooperation is in case when there exists a buffer at any line producing the foil. The lines can produce foil and store the bales into their buffers while the idle rolling lines can take the bales of foil from the buffers as they want.

A form of scheduling seems to be a more sophisticated strategy which makes an optimizing possible - e.g. minimizing the idle time.

However, it depends also on actual needs of foil parameters (especially on its width and thickness). The lines are not universal.

But the negotiation based on the offers and demands seems to be the most sophisticated strategy. The P/T PN-based modellig can be usable also on that way.

The example of negotiation of a couple of agents A_1, A_2.
Consider the following interpretation of P/T PN places:

p_1 - A_1 does not want to communicate;
p_2 - A_1 is available;
p_3 - A_1 wants to communicate;
p_4 - A_2 does not want to communicate;
p_5 - A_2 is available;
p_6 - A_2 wants to communicate;
p_7 - communication and
p_8 - availability of the communication by means of the interface (a communication channel).

The interpretation of the PN transitions is clear, but let us emphasize:

t_9 - fires the communication when A_1 is available and A_2 wants
t_{10} - fires the communication when A_2 is available and A_1 wants
t_{12} - fires the communication when both A1 and A2 wants to communicate each other.

The Negotiation of Agents

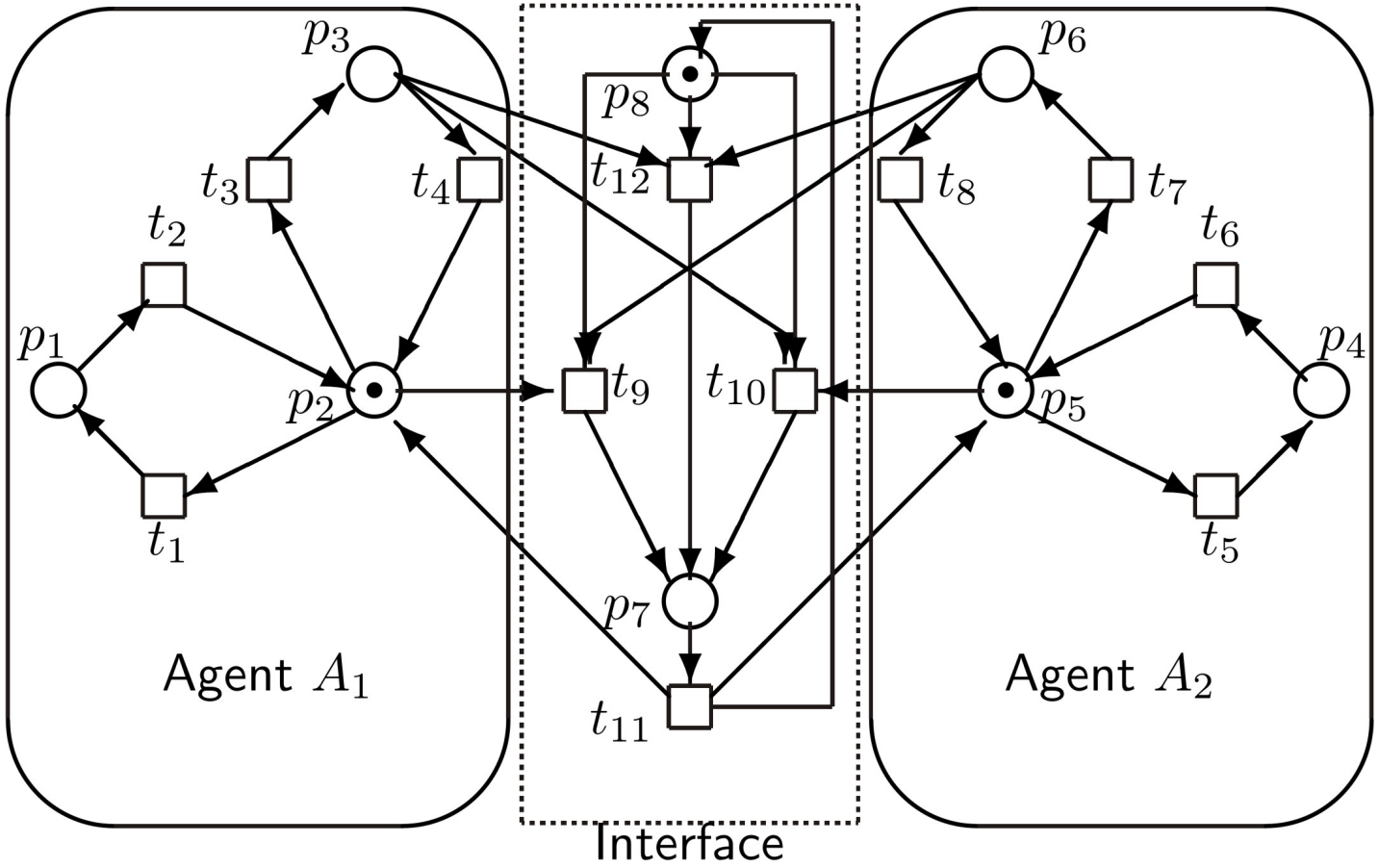

Figure 20. *The P/T PN model of the negotiation process*

From Fig. 20 it is clear, the interface realizing the negotiation process has the form of the PN module (PN subnet). The P/T PN based model of the agents communication has the following parameters

$$\mathbf{F} = \begin{pmatrix} \mathbf{F}_{A_1} & \mathbf{0} & \mathbf{F}_{A_1-A_2} \\ \mathbf{0} & \mathbf{F}_{A_2} & \mathbf{F}_{A_2-A_1} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_{neg} \end{pmatrix} = \left(\begin{array}{cccc|cccc|cccc} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{array}\right)$$

$$\mathbf{G}^T = \begin{pmatrix} \mathbf{G}_{A_1}^T & \mathbf{0} & \mathbf{G}_{A_1-A_2}^T \\ \mathbf{0} & \mathbf{G}_{A_2}^T & \mathbf{G}_{A_2-A_1}^T \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_{neg}^T \end{pmatrix} = \left(\begin{array}{cccc|cccc|cccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{array}\right)$$

To use the parameters at simulation it is necessary to choose an initial state.

Modelling of more cooperating agents in such a way is possible too.

Consider six production lines in a factory recycling the collected waste plastic.

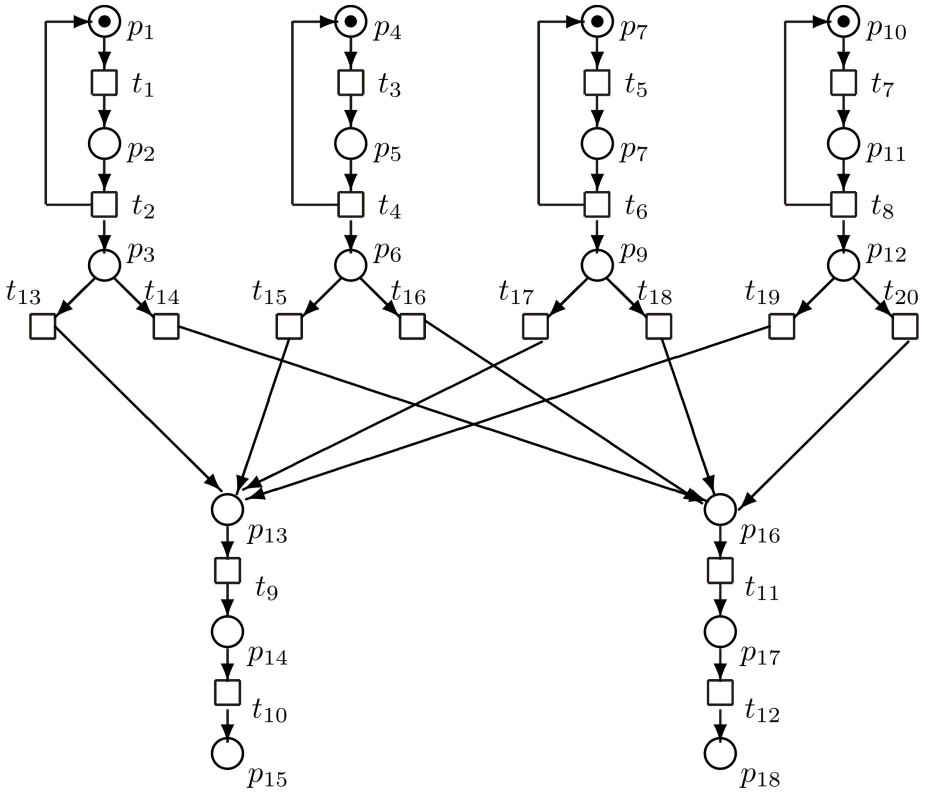

Figure 21. *The Petri net-based model of the rough conception of the supposed cooperation of the production lines*

Four upper lines produce the plastic double foil from the granulate prepared from the waste plastic. The FOHPN model of such a production line was presented above. Here, only the cooperation of the lines will be discussed.

Two lower lines produce rolls of plastic bags from the double foil. In Fig. 21 only a rough schema of the cooperation of two groups of lines is displayed

$\{p_1, p_4, p_7, p_{10}\}$ represent the continuous production of the foil

$\{p_2, p_5, p_8, p_{11}\}$ represent the cutting a bales of the foil with a determined weight and delivering the bale to the buffers $\{p_3, p_6, p_9, p_{12}\}$

$\{p_{13}, p_{16}\}$ represent the continuous rolling lines processing the double foil into the form of the belt of bags

$\{p_{14}, p_{17}\}$ represent the rolling the belt into rolls of a prescribed length (prescribed number of bags) and

$\{p_{15}, p_{18}\}$ represent buffers of the rolls

At forming the rules defining the mutual cooperation of the lines we have to respect the facts as follows:

(i) any bale of the foil from output buffers of the four foil production lines can enter only one of the two rolling machines;

(ii) only one bale can enter any rolling machine;

(iii) next bale can enter the rolling machines after finishing the rolling process

While (i), (ii) mean that the transition functions of the PN transitions $t_{13} - t_{20}$ has to satisfy

$$\gamma_{t_{13}} + \gamma_{t_{15}} + \gamma_{t_{17}} + \gamma_{t_{19}} \leq 1 \tag{2}$$

$$\gamma_{t_{14}} + \gamma_{t_{16}} + \gamma_{t_{18}} + \gamma_{t_{20}} \leq 1 \tag{3}$$

(iii) means that the places $p_{13} - p_{16}$ has to meet

$$\sigma_{p_{13}} + \sigma_{p_{14}} \leq 1 \tag{4}$$

$$\sigma_{p_{16}} + \sigma_{p_{17}} \leq 1 \tag{5}$$

The conditions can be satisfied by means of the supervision theory for DES.
Hence, the supervisor with the structure

$$\mathbf{F}_s = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

$$\mathbf{G}_s^T = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

and the initial state

$$^s\mathbf{x}_0 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

can be found.
Then, the PN model of cooperating lines is given in Fig. 22.

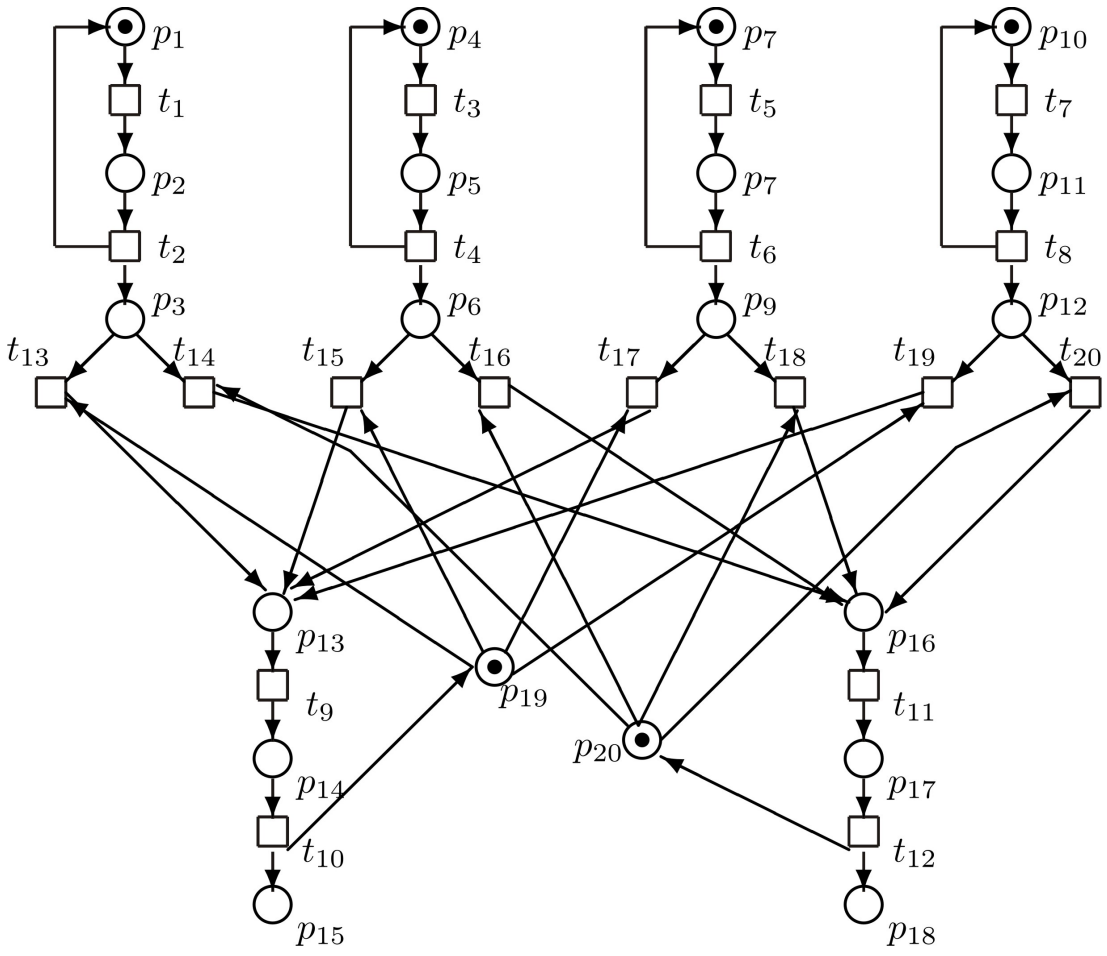

Figure 22. *The Petri net-based model of the supervised cooperation of the production lines*

Conclusion

P/T PN-based approach to agent modelling and cooperation was presented.

Agents, groups of agents and MAS are understood to be DES. Hence, P/T PN used for modelling, analyzing and control of DES are utilized here.

The modularity and supervison are the basic priciples of the presented approach.

The particular PN modules describe the autonomous agents and the supervision is utilized for synthesizing the agents cooperation.

The strategy of cooperation is determined by the aim of the global system (group of agents, MAS). It is prescribed in the form of conditions imposed on the global system in order to ensure the desired behaviour of it.

... Conclusion

When the conditions are given verbally, they have to be transformed into the system of inequations.

Introducing additional variables (slacks) into the inequations the system of equations is obtained. The slacks create the kernel of the supervisor to be synthesized.

Two kinds of supervision were presented - the supervision by means of PN P-invariants and that based on more general principle (utilizing PN places, transitions and the Parikh's vector)

Each of the approaches was illustrated by several examples.

Moreover, the hybrid agents were defined and the approach to synthesis of their cooperation in models of manufacturing systems was pointed out.

... Conclusion

The individual hybrid agents were modelled by means of FOHPN while the cooperation of the agents was modelled by P/T PN.

The instance of the concrete practical application - the recycling line producing the plastic double foil - was introduced in order to underline soundness of the approach.

By means of simulating the production line dynamic behaviour in Matlab the satisfying applicable results were obtained.

Thanks

Thank you very much
for your attention!!!