

BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume 14, No 1

Sofia • 2014

Print ISSN: 1311-9702; Online ISSN: 1314-4081 DOI: 10.2478/cait-2014-0004

Cooperation of Agents in Complex Systems Based on Supervision

František Čapkovič

Institute of Informatics, Slovak Academy of Sciences Dúbravská cesta 9, 845 07 Bratislava, Slovakia

Abstract: Complex systems consist of many cooperating devices. To have a transparent view on the system structure, as well as on the structural interconnections and cooperation of the subsystems, it is useful to synthesize the complex systems systematically in a prescribed order, even in analytical terms (if possible). The supervision of the subsystems seems to be a very suitable approach to accomplish these demands, and consequently it makes the complex systems diagnostics easier. The substantial agents (i.e., the agents of material nature -e.g., devices like particular production lines, robots, numerically controlled machines, etc.) can be coordinated and forced to cooperation by means of efficiently synthesized supervisors. The cooperation process has the character of DES (Discrete-Event Systems), because any system (including continuous systems), has minimally two discrete states - idle and working. DES control theory can be successfully utilized in supervisor synthesis. There are several approaches to modeling the agents and the process of supervisor synthesis. The Petri net-based approach is one of them. Place/Transition Petri Nets (P/T PN) are used here for modeling the behaviour of particular agents, as well as in the computational process used for the supervisor synthesis. Two main methods of the P/T PN-based supervision will be used, namely (i) the supervision based on the place invariants (P-invariants) of P/T PN, utilizing only the state vector during the supervisor synthesis, and (ii) the extended supervision utilizing not only the state vector, but also the control vector and Parikh's vector. The efficiency of the proposed approach is illustrated in a case study.

Keywords: Agent, complex systems, cooperation, supervision, supervisor synthesis.

1. Introduction

Aristotle (384-322 B.C.E.) was the author of the sentence: "The whole is more than the sum of its parts". Really, people working together can often accomplish tasks that could not be done working separately. The same is valid also for cooperating technical devices. Especially, the synergy of the subsystems in complex systems and the emergent behaviour of the complex system and/or Multi-Agent Systems (MAS) as a whole are in the centre of interest in the present research in different branches of science. The actual methods of cybernetics along with those of system theory, control theory and informatics are utilized in order to deal with the complex system design, analysis, control and diagnosis. Such an interdisciplinary approach can be also used for agents of different kinds of MAS - not only for software agents, social agents, etc., but also for different kinds of substantial (i.e., material) agents, e.g. robots, devices in manufacturing systems (like production lines, machine tools, automatically guided vehicles, etc.). The abstract systems based on Petri nets (PN) and digraphs are used in mathematical modeling of the subsystems and/or elementary agents and groups of them, as well as in analyzing their behaviour. The cooperation of subsystems/agents has the character of Discrete-Event Systems (DES). Consequently, DES control theory is useful at synthesizing supervisors realizing the agent cooperation. Even surroundings of technical complexes can be understood as a form of the cooperating agent. Agents are (Fonseca et al. [6]) persistent (software, but not only software) entities that can perceive, reason, and act in their environment and communicate with other agents. Namely, MAS are usually apprehended as a composition of collaborative agents working in a shared environment. In such way the agents together perform a more complex functionality. The communication among the agents in MAS enables the agents to exchange information. Consequently, the agents can coordinate their actions and cooperate with each other. The agent behaviour has a character of DES, because it is the system driven by occurrence of discrete events. Namely, the agent persists in a given state (e.g., a kind of activity) until then when an occurrence of a discrete event forces it to change the state into another one (e.g., to finish or abort the previous activity and to start another one). The agent behaviour involves both internal and external attributes. While the external attributes are as given in Demazeau [5] that the agent (i) evolves in an environment; (ii) is able to perceive this environment; (iii) is able to act in this environment; (iv) is able to communicate with other agents; (v) exhibits an autonomous behaviour, the internal attributes of the agent are that it encompasses a local control in some of its perception, communication, knowledge acquisition, reasoning, decision, execution, and action processes. While the internal attributes characterize rather the agent inherent abilities, the external attributes of the agents are manifested themselves in different measures in a rather wide spectrum of MAS applications.

To describe the behaviour of DES, Petri nets Peterson [10], Murata [9] are frequently used. PN yield both the graphical model and the mathematical one, and they have formal semantics too. The so called Place/Transition PN (P/T PN) will be used in this paper. For simplicity, below we will use the abbreviation PN.

Because there exist many techniques for proving the PN basic properties (like reachability, liveness, boundness, conservativeness, reversibility, coverability, persistence, fairness and so one), PN represent the sufficient general means to be able to model a wide class of systems. Some of them are very useful also in diagnostics of the modeled complex systems. These arguments are usually used in order to prefer PN to other approaches. In addition, there were developed many methods in PN theory that are very useful at model checking, as well as in complex systems diagnostics - e.g., like the methods of the deadlocks avoidance, methods for computing P-invariants (i.e., place-invariants) and T-invariants (transitioninvariants), etc. Summing up, the modeling power of PN consists especially in the facts that (i) PN have formal semantics and consequently, the execution and simulation of PN models are unambiguous; (ii) the notation of modeling a system is event-based, i.e., PN can model both states and events; (iii) there are many analysis techniques associated with PN. Especially, the approach based on P/T PN enables us to use linear algebra and matrix calculus - exact, and verified approaches in practice. This makes possible the complex systems analysis in analytical terms, especially, by computing the states reachability tree (RT), invariants, testing properties, model checking, even the efficient model-based control synthesis. Mutual interactions of the agents are considered within the framework of the global model. Such an approach is sufficiently general in order to allow the design of the model that yields the possibility to analyze any situation. Even the environmental behaviour can be modeled as an agent of the agent system also. Thus, the model can acquire an arbitrary structure and generate different situations.

2. Modeling DES by Petri Nets

DES are systems discrete in nature – i.e., driven by occurrence of discrete events. Let us model DES by means of P/T PN. Use the analogy between the DES atomic activities $a_i \in \{a_1, \dots, a_n\}$ and the PN places $p_i \in \{p_1, \dots, p_n\}$, as well as between the discrete events $e_j \in \{e_1, \dots, e_m\}$, occurring in DES and the PN transitions $t_j \in \{t_1, \dots, t_m\}$. Then DES behaviour can be modeled by means of P/T PN. As for the structure, PN are bipartite directed graphs $\langle P, T, F, G \rangle$ with two sets of nodes namely, set of places P and set of transitions T and two kinds of edges namely, the set of directed arcs from places to transitions F and the set of directed arcs from transitions to places G. The transition function $\delta : X \times U \rightarrow X$, where X is the set of state vectors of PN places (i.e., the PN state vectors), while U is the set of the state vectors of PN transitions (i.e., the PN control vectors) represent the PN dynamics. The symbol δ points out the fact that a new state vector of PN depends on the existing states and occurrence of discrete events. Fortunately, it can be expressed in the form of a linear discrete system representing the analytical model of PN dynamics, having the form

(1)
$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{B} \cdot \mathbf{u}_k, \quad k = 0, \cdots, K, \quad \mathbf{B} = \mathbf{G}^{\mathrm{T}} - \mathbf{F},$$

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

Here, *k* is the discrete step of the dynamics development; $\mathbf{x}_k = (\sigma_{p_1}^k, \dots, \sigma_{p_n}^k)^T$ is the *n*-dimensional state vector at step *k* with $\sigma_{p_i}^k \in \{0, 1, \dots, c_{p_i}\}, i = 1, \dots, n$, expressing the states of PN places (DES atomic activities); $\sigma_{p_i}^k = 0$ expresses the passivity, while the activity is expressed by $0 \le \sigma_{p_i}^k \le c_{p_i}$ with c_{p_i} being the capacity (as to the number of marks in p_i); the passivity means, e.g. an empty buffer, while the activity means a number of parts stored in the buffer and the capacity is understood to be the maximal number of parts which can be put into the buffer; $\mathbf{u}_k = (\gamma_{t_1}^k, \dots, \gamma_{t_m}^k)^T$ is the *m*-dimensional control vector of the system at step *k* with components $\gamma_{t_j}^k \in \{0, 1\}, j = 1, \dots, m$; they represent the occurrence of failures, etc.); when *j*-th discrete event is enabled, $\gamma_{t_j}^k = 1$, while $\gamma_{t_j}^k = 0$ when the event is disabled; **F**, **G** are incidence matrices of the directed arcs corresponding, respectively, to the sets *F*, *G*; (.)^T symbolizes the matrix or vector transposition.

3. Supervision in agent cooperation

(2)

The modular approach in building a complex system makes possible to model and analyze each module separately, as well as the global composition of modules. In general, three principal different kinds of the model creation $\check{C} a p k o v i \check{c}$ [1]; $\check{C} a p k o v i \check{c}$ and J o t s o v [4] can be distinguished according to the form of the interface connecting the modules (PN subnets), namely (i) the interface consisting exclusively of PN transitions; (ii) the interface consisting exclusively of PN places; (iii) the interface in the form of a PN subnet with an arbitrary structure containing both positions and transitions.

Let us deal with the cooperation of agents A_i , $i = 1, ..., N_A$, by means of the deeper places. Here, the synthesis in analytical terms by means of the PN-based approach is possible. Usually, the material agents themselves are not able to coalesce on a procedure, satisfying all of them because the autonomous agents are usually egoistic (selfish). Violent driving of the individual agents in the limited space (restricted area) might tend to wrecks with exterminatory effects, including even some mechanical devastations and standing the global complex system off. Therefore, the supervisor determines a policy of the agents behaviour from a global point of view (i.e., conductive to the whole complex), in order to achieve satisfying results of the cooperative interaction among devices and the expected behaviour of the whole complex. However, in the same time it guarantees that no agent will be discriminated in its activities.

Let us regard the supervisor synthesis process from the opposite side. The agent cooperation strategy of the supervisor evokes an impression that such a

process expresses the agent negotiation (although unwilling by the autonomous agents themselves). Such a view is not any fantasy. Namely, the supervisor does not exhibit any own self-interest. Its activity is only focused on the realization of the global objective function (a criterion dictating the behaviour) of the complex system. In other words, the supervisor only forces demands on the behaviour of agents (conducive to the global goal of the complex system) and ensures their realization. In case of realizing the agent cooperation through the PN places, the supervisor can be synthesized in analytical terms by virtue of the prescribed conditions.

3.1. Supervision based on P-invariants of PN

The first method to synthesize the supervisor will utilize the approach based on Pinvariants of P/T PN method (I or d a c h e and A n t s a k l i s [7]; Č a p k o v i č [2]). P-invariants are vectors **v**, with the property that multiplication of these vectors by any state vector **x** reachable from a given initial state vector \mathbf{x}_0 yields the same result (the relation of the state conservation)

$$\mathbf{v}^{\mathrm{T}}.\mathbf{x} = \mathbf{v}^{\mathrm{T}}.\mathbf{x}_{0}$$

Taking into account the consecutive states (that are obtained by firing of only one transition), it results that

(4)
$$\mathbf{v}^{\mathrm{T}}.\mathrm{col}_{t}(\mathbf{B}) = 0,$$

for each transition *t*. Here, $col_t(\mathbf{B})$ represents the column of **B** corresponding to the transition *t*. It means that, algebraically, these vectors are solutions of the following equation

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

with **v** being an *n*-dimensional vector and **0** being the *m*-dimensional zero vector, which is usually introduced – see, e.g., Murata [9] – as the definition of the P-invariant of PN. However, there are usually several P-invariants in PN models. Hence, the set of the P-invariants of PN is created by the columns of the $(n \times n_s)$ dimensional matrix **V** (n_s expresses the number of invariants), being the solution of the equation as follows:

$$\mathbf{V}^{\mathrm{T}}.\mathbf{B}=\mathbf{0}.$$

The main idea of the approach to supervisor synthesis consists in the following. Let us prescribe the conditions for linear combinations of entries of the state vector \mathbf{x} . In a matrix form it is as follows:

$$\mathbf{L}_{\mathbf{p}} \cdot \mathbf{X} \le \mathbf{b}$$

where \mathbf{L}_p is the $(n_s \times n)$ -dimensional matrix of integers (where nonzero entries represent the multiplicity of participation of the corresponding entries of the state vector \mathbf{x} under the conditions in question) and \mathbf{b} is the n_s -dimensional vector of integers representing constants (representing the maximal number of marks inside the PN places kept together by the places participating in the condition in question). Imbedding of additional PN places (slacks) into the inequality condition (7) we can transform the inequality condition into the following equation:

(6)

$$\mathbf{L}_{p} \cdot \mathbf{x} + x_{s} = \mathbf{b}$$
.

Hence, we can find the structural interconnections between them and the original PN places. Extend (6) into the form

(9)
$$(\mathbf{L}_{p} \quad \mathbf{I}_{s}) \cdot \begin{pmatrix} \mathbf{B} \\ \mathbf{B}_{s} \end{pmatrix} = \mathbf{0}$$

we force P-invariants to the supervised system. Here \mathbf{I}_s is a $(s \times s)$ -dimensional identity matrix and \mathbf{B}_s is the supervisor structure (till now still unknown) which is searched by the synthesis process. Consequently, a new PN subnet representing the supervisor and its interface with the original system are found and added to the PN model. Namely, after multiplying the matrices in (9), we have

(10)
$$\mathbf{L}_{p}.\mathbf{B} + \mathbf{B}_{s} = \mathbf{0}$$
, and thus $\mathbf{B}_{s} = -\mathbf{L}_{p}.\mathbf{B}$

The initial state of the supervisor can be found from (8) as follows:

(11)
$$\mathbf{x}_{s}^{0} = \mathbf{b} - \mathbf{L} \cdot \mathbf{x}_{0}.$$

The matrix \mathbf{B}_s can be decomposed (factorized) into the matrices $\mathbf{F}_s, \mathbf{G}_s^{\mathrm{T}}$, namely $\mathbf{B}_s = \mathbf{G}_s^{\mathrm{T}} - \mathbf{F}_s$. These matrices represent the supervisor structure, more precisely the interconnections of the incorporated slacks with the original PN structure. The state vector and the incidence matrices of the augmented system (i.e., the original system together with the supervisor), are the following:

(12)
$$\mathbf{x}_{a} = \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_{s} \end{pmatrix}; \quad \mathbf{F}_{a} = \begin{pmatrix} \mathbf{F} \\ \mathbf{F}_{s} \end{pmatrix}; \quad \mathbf{G}_{a}^{\mathrm{T}} = \begin{pmatrix} \mathbf{G}^{\mathrm{T}} \\ \mathbf{G}_{s}^{\mathrm{T}} \end{pmatrix}.$$

3.2. Supervision by means of the extended method

Although the supervisor synthesized by means of a P-invariant method can cover a wide class of conditions, the method defined by I or d a c h e [8]; Č a p k o v i č [1] extends the class of the condition to a considerable extent. Namely, the condition has the form

(13)
$$\mathbf{L}_{\mathbf{p}}.\mathbf{x} + \mathbf{L}_{t}.\mathbf{u} + \mathbf{L}_{v}.\mathbf{v}_{\mathbf{p}} \leq \mathbf{b},$$

where \mathbf{L}_t and \mathbf{L}_v are $(n_s \times m)$ -dimensional -dimensional integer matrices and \mathbf{v}_p is *m*-dimensional integer vector named as Parikh's vector. \mathbf{L}_t and \mathbf{L}_v have analogical meaning as \mathbf{L}_p , however they concern the linear combinations of the control vector entries and the Parikh vector entries, respectively. The Parikh vector is obtained by the consecutive development of the system (1)

(14)
$$\mathbf{x}_{1} = \mathbf{x}_{0} + \mathbf{B} \cdot \mathbf{u}_{0}; \ \mathbf{x}_{2} = \mathbf{x}_{1} + \mathbf{B} \cdot \mathbf{u}_{1} = \mathbf{x}_{0} + \mathbf{B} \cdot (\mathbf{u}_{0} + \mathbf{u}_{1}); \ \dots;$$
$$\mathbf{x}_{k} = \mathbf{x}_{0} + \mathbf{B} \cdot (\mathbf{u}_{0} + \mathbf{u}_{1} + \dots + \mathbf{u}_{k-1}).$$

 $\mathbf{x}_{k} = \mathbf{x}_{0} + \mathbf{B} \cdot (\mathbf{u}_{0} + \mathbf{u}_{1} + \ldots + \mathbf{u}_{k-1}).$ It means that $\mathbf{x}_{k} = \mathbf{x}_{0} + \mathbf{B} \cdot \sum_{j=0}^{k-1} \mathbf{u}_{j}$. Hence, the Parikh's vector is just the vector

(15)
$$\mathbf{v}_{p} = \sum_{j=0}^{k-1} \mathbf{u}_{j}.$$

45

(8)

Its entries give us important information about how many times the particular transitions are fired during the development of the P/T PN dynamics from the initial state \mathbf{x}_0 to a final (terminal) state \mathbf{x}_k . When $\mathbf{b} - \mathbf{L}_p \cdot \mathbf{x} \ge \mathbf{0}$ holds, the supervisor structure is as follows:

(16)
$$\mathbf{F}_{s} = \max\left(\mathbf{0}, \mathbf{L}_{p}, \mathbf{B} + \mathbf{L}_{v}, \mathbf{L}_{t}\right),$$

(17)
$$\mathbf{G}_{s}^{\mathrm{T}} = \max\left(\mathbf{0}, \mathbf{L}_{t} - \max\left(\mathbf{0}, \mathbf{L}_{p}, \mathbf{B} + \mathbf{L}_{v}\right)\right) - \min\left(\mathbf{0}, \mathbf{L}_{p}, \mathbf{B} + \mathbf{L}_{v}\right).$$

Its initial state is given in the form

(18)
$$\mathbf{x}_{s}^{0} = \mathbf{b} - \mathbf{L}_{p} \cdot \mathbf{x}_{0} - \mathbf{L}_{v} \cdot \mathbf{v}_{p}^{0},$$

where \mathbf{v}_{p}^{0} is the initial Parikh's vector. The operators max (.) and min (.) are, respectively, the operators of maximum and minimum for matrices. They are executed element by element.

4. Case study – illustrative example

Consider the machine of a manufacturing system cooperating with two automatically guided vehicles. AGV1 transports correct products to a buffer, while AGV2 transports the bad products to another buffer in case the machine fails. The PN-models of the autonomous agents - the devices Machine, Transport 1 and Transport 2 are displayed in Fig. 1a. Here, the interpretation of the PN places and transitions is as follows: p_1 – a part is being carried to completed-parts queue by AGV1; p_2 – AGV1 is free; p_3 – AGV1 is at pick-up position at machine M; p_4 – a part is being carried to the damaged-parts queue; $p_5 - AGV2$ is free; $p_6 - AGV2$ is at a pick-up position in machine M; $p_7 - M$ is up and busy (a part is being processed); $p_8 - M$ is free; $p_9 - M$ is out of order and it is being repaired; p_{10} - the completed part is waiting for transfer; p_{11} – the damaged part is waiting for transfer; t_1 – the part is picked up by AGV1; t_2 – the part is deposited in a completed-parts queue by AGV1; t_3 – AGV1 moves at a pick-up position in the machine; t_4 – the part is picked up by AGV2; t_5 – the part is deposited in the damaged-parts queue by AGV2; t_6 – AGV2 moves at the pick-up position at M; t_7 – uncontrollable: processing of the part is complete; t_8 – the part is charged in M; t_9 – uncontrollable: the machine fails, the part is damaged; $t_{10} - M$ is repaired.

The structure of the P/T PN-based models of the autonomous agents – machine M, Transport 1 by AGV1, and Transport 2 by AGV2, are the following:

(19)
$$\mathbf{F}_{\mathrm{M}} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}; \quad \mathbf{G}_{\mathrm{M}}^{\mathrm{T}} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}; \quad \mathbf{x}_{\mathrm{M}}^{0} = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^{\mathrm{T}}.$$



Fig. 1. The autonomous agents (a) and the structure after the first step of supervision

(20)
$$\mathbf{F}_1 = \mathbf{F}_2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}; \quad \mathbf{G}_1^{\mathrm{T}} = \mathbf{G}_2^{\mathrm{T}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}; \quad \mathbf{x}_1^0 = \mathbf{x}_2^0 = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^{\mathrm{T}}.$$

Consequently, we have the system of autonomous agents with parameters

(21)
$$\mathbf{F} = \begin{pmatrix} \mathbf{F}_{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_{M} \end{pmatrix}; \quad \mathbf{G}^{\mathrm{T}} = \begin{pmatrix} \mathbf{G}_{1}^{\mathrm{T}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{2}^{\mathrm{T}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_{M}^{\mathrm{T}} \end{pmatrix}; \quad \mathbf{x}_{0} = \begin{pmatrix} \mathbf{x}_{0}^{0} \\ \mathbf{x}_{2}^{0} \\ \mathbf{x}_{M}^{0} \end{pmatrix}.$$

4.1. The first step of the supervision

To satisfy the global system technology, the transport of the correct parts produced by M is realized by means of AGV1 in Transport 1 and the transport of the bad parts produced by M is realized by AGV2 in Transport 2. Hence, the corresponding priorities can be defined by the conditions imposed on Parikh's vector entries as follows:

(22)
$$v_1 \le v_7 \text{ and } v_4 \le v_9$$

Namely, from Fig. 1a it is clear that the parts have to be produced before their transport to the buffer. Thus, in condition (13) only matrix \mathbf{L}_{ν} will be nonzero, namely, in the following form:

(23)
$$\mathbf{L}_{\nu} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \end{pmatrix}; \quad \mathbf{b} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

(24)
$$\mathbf{v}_{p}^{0} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^{1}$$
.

Consequently, in virtue of (16), (17) we obtain supervisor S_1 with the following parameters

Supervisor S_1 is created by the PN places p_{10} , p_{11} together with the directed arcs connecting it with the agents. It is clear that S_1 is not sufficient yet. Namely, such a configuration has 267 reachable states and, moreover, the return to the initial state in order to realize the next working cycle is impossible. Such a model is insufficient for practical usage. Consequently, the next step of the synthesis is necessary. Therefore, we have to perform the next step of the supervision. After the first step we have the structure (an augmented system consisting of autonomous agents and supervisor S_1), given in Fig. 1b. Namely,

(27)
$$\mathbf{F}_{\mathrm{I}} = \begin{pmatrix} \mathbf{F} \\ \mathbf{F}_{\mathrm{S}_{\mathrm{I}}} \end{pmatrix}; \quad \mathbf{G}_{\mathrm{I}}^{\mathrm{T}} = \begin{pmatrix} \mathbf{G}^{\mathrm{T}} \\ \mathbf{G}_{\mathrm{S}_{\mathrm{I}}}^{\mathrm{T}} \end{pmatrix}; \quad \mathbf{x}_{\mathrm{I}}^{0} = \begin{pmatrix} \mathbf{x}_{0} \\ \mathbf{x}_{\mathrm{S}_{\mathrm{I}}}^{0} \end{pmatrix}$$

In the next step we will start from structure (27).

4.2. The second step of the supervision

Here we will use the P-invariant based approach to the supervision. Now only matrix L_p in the general condition (13) will be nonzero. The aim of this step of the supervision is to ensure the possibility to reach the initial state (to realize the working cycle) and to find the satisfying throughput in order to reduce the number of states and especially the number of possible state trajectories. Hence, the following conditions will be imposed on the system (27):

(28)
$$\sigma_{p_3} + \sigma_{p_6} \le 1 \text{ and } \sigma_{p_7} + \sigma_{p_{10}} + \sigma_{p_{11}} \le 1.$$

The first of them eliminates the simultaneous access of AGV1 and AGV2 to M, while the second one expresses the situation, that a part is either produced by M or transported by AGV1 to the buffer of correct parts, or transported by AGV2 to the buffer of bad parts. Hence,

(30)
$$\mathbf{x}_{I}^{0} = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}^{\mathrm{T}}$$

Using (10), (11) we obtain the parameters of the second supervisor S_2 as follows

Hence, the structure of the augmented system is

(33)
$$\mathbf{F}_{\mathrm{II}} = \begin{pmatrix} \mathbf{F}_{\mathrm{I}} \\ \mathbf{F}_{\mathrm{S}_{2}} \end{pmatrix}; \quad \mathbf{G}_{\mathrm{II}}^{\mathrm{T}} = \begin{pmatrix} \mathbf{G}_{\mathrm{I}}^{\mathrm{T}} \\ \mathbf{G}_{\mathrm{S}_{2}}^{\mathrm{T}} \end{pmatrix}; \quad \mathbf{x}_{\mathrm{II}}^{\mathrm{0}} = \begin{pmatrix} \mathbf{x}_{\mathrm{I}}^{\mathrm{0}} \\ \mathbf{x}_{\mathrm{S}_{2}}^{\mathrm{0}} \end{pmatrix}.$$

The supervisor S_2 is created by places p_{12} , p_{13} together with the directed arcs connecting them with the previous structure (27). In such a way the number of states was strongly reduced to 48 and the initial state became reachable. The structure is displayed in Fig. 2a. However, testing the properties of the PN structure (33) by means of a reachability graph it was found that now there are two deadlocks here, namely, in the states

These two states are reachable, respectively, by the transition sequences $\{t_6, t_8, t_7\}$ and $\{t_3, t_8, t_9, t_{10}\}$. Consequently, we can use the next step of the synthesis to avoid the problem with the deadlocks.

4.3. The third step of the supervision

In order to remove the deadlocks we have to start from (33) and strictly define the priorities concerning the departure of AGV1, AGV2 from machine M with the parts to the buffers against their arrival to machine M, in order to take the parts from the machine. The following conditions have to be imposed on the system (33)

$$v_3 \le v_7 \quad \text{and} \quad v_6 \le v_9.$$

The first one means that the transport of finished part by AGV1 to the buffer has a priority against the arrival of empty AGV1 towards the machine. The same refers to the second inequality in (36) for AGV2.



Fig. 2. The structure after the second step of the supervision (a), and the structure after the third step of supervision (b)

(37)
$$\mathbf{L}_{v} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \end{pmatrix}; \quad \mathbf{b} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

(38)
$$\mathbf{v}_{p}^{0} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^{T}$$
.

Thus,

(40) $\mathbf{G}_{\mathbf{S}_{3}}^{\mathrm{T}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}; \quad \mathbf{x}_{\mathbf{S}_{3}}^{0} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$

Hence, the augmented system is as follows:

(41)
$$\mathbf{F}_{\mathrm{III}} = \begin{pmatrix} \mathbf{F}_{\mathrm{II}} \\ \mathbf{F}_{\mathrm{S}_{3}} \end{pmatrix}; \quad \mathbf{G}_{\mathrm{III}}^{\mathrm{T}} = \begin{pmatrix} \mathbf{G}_{\mathrm{II}}^{\mathrm{T}} \\ \mathbf{G}_{\mathrm{S}_{3}}^{\mathrm{T}} \end{pmatrix}; \quad \mathbf{x}_{\mathrm{III}}^{\mathrm{0}} = \begin{pmatrix} \mathbf{x}_{\mathrm{II}}^{\mathrm{0}} \\ \mathbf{x}_{\mathrm{S}_{3}}^{\mathrm{0}} \end{pmatrix}.$$

The supervisor S_3 is created by places p_{14} , p_{15} together with their interconnections with the previous structure (33). The number of states was reduced to 30 and the deadlocks were removed. The structure is displayed in Fig. 2b.

4.4. The other possible steps of the supervision

In general, the other improvements are possible, of course, under the condition that we are able to form constraints bringing an asset. In this case the procedure can be realized analogically. However, in our case study it is practically impossible. In cases where improvements are possible, inequality (13) can be utilized in seven modifications. Namely, the actual step of the supervision can be applied in the form given in (13), i.e., all of the three matrices are nonzero (it is the first modification). The other three modifications are represented by the situation when one of the matrix is missing putting it to be equal to zero and only two matrices are nonzero – i.e., there are three possible combinations. Finally, other three possibilities arise when only one matrix is nonzero and two others are zero. It gives a very wide spectrum of conditions in the process of supervision.

5. Conclusion

An alternative view on the synthesis of the agent cooperation in order to control complex systems was introduced in this paper. The modular approach starting from models of autonomous devices (substantial agents) towards their mutual interactive cooperation was proposed by means of consecutive steps of supervision. In this way we are able to come successively to the final result – i.e., to desired and satisfying behaviour of the global complex systems. These facts are very important also for complex system diagnostics. Namely, the more detailed the process of synthesis of the complex system dynamic behaviour is, the simpler is the diagnostics process.

The approach was illustrated in a case study in details. The proposed modular approach seems sufficiently general to be used also in the composition of different synthesized fragments into a larger and complex system with a more complicated structure among particular agents and/or subsystems. Thus, there exists a motivation for further research.

Acknowledgment: The work was partially supported by the Slovak Grant Agency for Science VEGA under Grant No 2/0039/13 (2013-2016) and the contract between the Institute of Informatics of the Slovak Academy of Sciences and the Institute of Information and Communication Technologies of the Bulgarian Academy of Sciences "Agent Oriented Diagnostics of Complex Systems" (2012-2014). The author thanks VEGA, as well as SAS and BAS for the support.

References

- Č a p k o v i č, F. A System Approach to Describing and Analysing the Behaviour of Agents in MAS. – In: Austrian Society for Cybernetics Studies, Cybernetics and Systems, Vol. 1, 2008, 70-75. R. Trappl, Ed., Vienna, Austria.
- Č a p k o v i č, F. Automatic Control Synthesis for Agerts and Their Cooperation in MAS. Computing and Informatics, Vol. 29(6+), 2010, 1045-1071.
- Č a p k o v i č, F. Cooperation of Agents in Manufacturing Systems. In: P. Jedrzejowicz, N. T. Nguyen, R. J. Howlett, C. J. Lakhmi, Eds. Agent and Multi-Agent Systems: Technologies and Applications. Lecture Notes on Artificial Intelligence, Vol. 6070, Berlin-Heidelberg, Germany, Springer-Verlag, 2010, 193-202.
- Č a p k o v i č, F., V. J o t s o v. A System Approach to Agent Negotiation and Learning. In: V. Sgurev, M. Hadjiski, J. Kacprzyk, Eds. Intelligent Systems: From Theory to Practice. Series: Studies in Computational Intelligence, Vol. 299, Berlin-Heidelberg, Germany, Springer-Verlag, 2010, 133-160.
- D e m a z e a u, Y. MAS Methodology. Tutorial at 2nd French-Mexican School of the Repartee Cooperative Systems. – In: ESRC 2003, September 29-October 4 2003. Rennes, France: IRISA.
- 6. F o n s e c a, S., M. G r i s s, R. L e t s i n g e r. Agent Behavior Architectures A MAS Framework Comparison (HP Labs Technical Report HPL-2001-332), 2001. Palo Alto, USA: HP.
- 7. I o r d a c h e, M. V., P. J. A n t s a k l i s. Supervisory Control of Concurrent Systems: A Petri Net Structural Approach. Boston, MA, USA, Birkhäuser, 2006.
- I or d a c h e, M. V. Methods for the Supervisory Control of Concurrent Systems Based on Petri Nets Abstraction. Ph.D. Dissertation. University of Notre Dame. Notre Dame, Indiana, USA, 2003.
- M u r a t a, T. Petri Nets: Properties, Analysis and Applications. Proceedings of IEEE, Vol. 77, 1989, No 4, 541-588.
- Peterson, J. L. Petri Net Theory and Modeling the Systems. New Jersey, Englewood Cliffs, USA, Prentice Hall, Inc., 1981.