Journal of Business Economics and Management 2008
9(3): 219–226

SIMULATION MODELLING CONSTRUCTION PROJECT WITH REPETITIVE TASKS USING PETRI NETS THEORY

Sławomir Biruk[1], Piotr Jaśkowski[2]

Lublin University of Technology,
ul. Nadbystrzycka 40, 20-618 Lublin, Poland
E-mails: [1] s.biruk@pollub.pl, [2] p.jaskowski@pollub.pl

Received 5 May 2007; accepted 20 March 2008

Abstract. Resources selection and allocation at the project planning stage is an important issue for reducing project cost, duration and risk. Existing planning and scheduling methods overlook aspects of limited production capacity of construction companies (contractors) due to the fact that they are simultaneously engaged in realization of some projects. This paper presents a new methodology for project scheduling with repetitive processes using Petri nets based approach. The paper starts with an overview of current developments in the Petri nets theory. We then propose an efficient computational method based on simulation of Petri net model for construction project planning and subcontractor agreement analysis. An example of construction project simulation research is presented to illustrate the method of project planning and resources allocation.

Keywords: project planning, construction project scheduling, repetitive processes scheduling, resources allocation, Petri nets, simulation modelling.

1. Introduction

Petri nets are an effective tool for modelling discrete event systems. Their essential advantage is the possibility of mapping concurrency, asynchronism, hierarchism of modelled systems. Due to these advantages Petri nets are used for modelling and analysing complex discrete event dynamic systems such as manufacturing, operational systems, distributed data bases, and also for complex construction processes; see Körner and Franz (1989), Sawhney, Mund, and Marble (1999), Sawhney, Abudayych, and Chaitavatputtiporn (1999), Sawhney (1997), Wakefield and Sears (1997) for examples.

The classic methodology of CPM/PERT, used for construction project scheduling is based on Berg's class graphs, which are a subclass of Petri nets called marked graphs, or, to be more specific, event graphs. Contrary to CPM/PERT method, Petri nets can include cycles what simplifies the manner of repetitive processes modelling. Application of different probabilistic rules to resources allocation (FIFO, LIFO etc.) and the possibility of using different logic conditions to start some tasks make Petri nets a powerful tool, similar to queuing networks. The dynamics of systems modelled by Petri nets is described by a state vector variable in time (variable number of tokens in particular places). State vector represents the temporary lengths of queues. The same features enable modelling systems of probabilistic structure (like in Generalized Activity Networks).

Petri nets properties such as safeness, boundedness, conservativeness, liveness, reachability and methods of their study have been discussed e.g. by DiCesare *et al.* (1993) and Reisig (1982).

2. Nets in construction

The legible graphic representation of Petri nets comprising few symbols make the method a useful tool for mapping complex conditions in construction. It facilitates modelling and designing construction processes of different complexity level (either particular processes or whole construction projects). The type of Petri nets (e.g. Condition/Event–systems, Place/Transition–nets (P/T–nets), nets with individual tokens: Predicate/Transition nets (PrT–nets) and Coloured Petri nets – CP–nets used for modelling are usually chosen according to the character of the analysed problem.

ISSN 1611-1699 print. *www.jbem.vgtu.lt*, *www.jbem.lt* DOI: 10.3846/1611-1699.2008.9.219-226

Wakefield and Sears (1997) applied P/T–nets to model a concrete placement operation. The same problem was analysed before by Halpin and Riggs (1992), who used Microcyclone method. Simulation results of both models were comparable. Wakefield and Sears claim that Petri nets are a more universal tool for modelling cyclical processes than Microcyclone method.

Sawhney, Mund, and Marble (1999) applied hierarchical modelling methodology to mapping a steel erection process. This approach has been developed and also used by Sawhney *et al.* (1999) to analyse a concrete production plant. The model consists, among others, of modules like: the control room process, the mixing and transportation process and the cement ordering process. The research aim was to determine daily plant productivity versus the number of concrete trucks. The authors used Jensen's concept (see Jensen 1994, 1997) of fusion places to represent shared resources by a number of work tasks, e.g. one crane is used for unloading precast elements, structure assembly and concrete placement.

Mapping of large and complex systems by means of P/T–nets is complicated, which impedes qualitative and quantitative analysis. Therefore, large systems with a great number of logical dependences between processes are modelled using Nets with Individual Tokens. These are Predicate/Transition nets (PrT–nets), described by DiCesare *et al.* (1993), Genrich (1991) and Reisig (1982), and nets with coloured tokens (CP–nets) presented by DiCesare (1993), Jensen (1991), Lin and Lee (1997). Transition firing causes a change of the number of tokens and (or) their type (colour). Predicates attributed to transitions, together with arc conditions, decide whether transition is enabled to fire and determine the change of net marking after firing this transition. Chang and Luh (1997) and Lin and Lee (1997) argue that net models with individual tokens are useful especially for modelling different types of resources involved.

Coloured Petri nets have been used, among others, by Wakefield and Sears (1997) to model earthworks at an airport construction site in Hong Kong. The use of various transporting routes and various kinds of trucks persuaded the authors to apply nets with individual tokens, probabilistic transition and queuing priorities in order to simplify the numerical model. The simulation experiment facilitated the choice of a set of machines optimal in terms of efficiency.

Predicate/Transitions nets of hierarchical structure have been used by Körner and Franz (1989) to analyse a concrete placement process. The model was analysed by a computer simulation method at multiple change of initial marking net (boundary conditions). As a result of the simulation experiment, productivity and utilisation ratio of particular construction machines were calculated. Simulation research was aimed at determination of bottlenecks of the production system and cost optimisation.

3. Place / Transition nets overview

Each PrT–net and CP–net used for modelling very complex systems with a large number of constraints and logical dependences among system elements can be translated to P/T net. Each net created according to a classical CPM/PERT method can be also presented in the form of P/T Petri net (see Fig. 1). Due to that fact, the same system (construction project) can be analysed using both methods. Horns (1989) and Magott and Skudlarski (1989) used CPM/PERT analysis for initial system assessment and Petri net was automatically generated and analysed afterwards.

The graphic representation of Petri net is a weighted-bipartite directed graph with two types of nodes. Elements $p \in P$ called places are drawn as circles. They represent conditions (completion of preceding activities and resources availability) necessary to start an activity. Elements $t \in T$ called transitions are shown as boxes or bars. They represent activities, which may consume time. Places and transitions are connected by directed arcs. Arcs can be labelled by a number called arcs weight. Places, transitions and arcs create a static net structure. Petri net is defined as a quadruple:

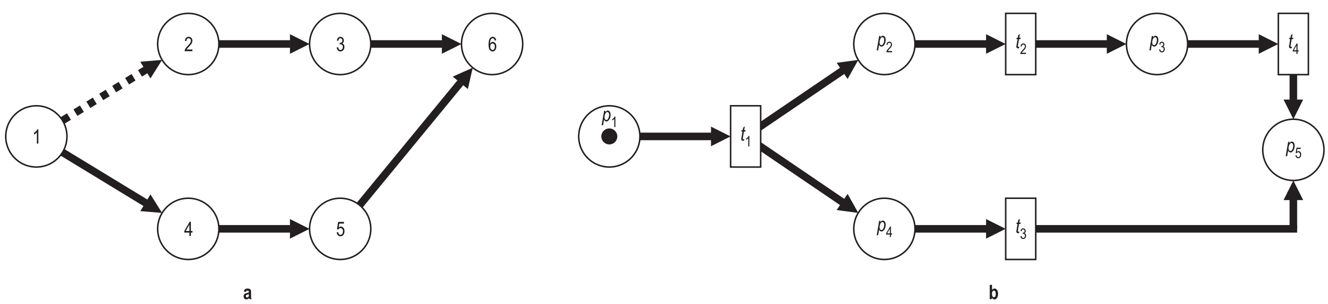

Fig. 1. Project network model: a) in CPM-PERT method, b) Petri net

$$N = \langle P, T, Pre, Post \rangle, \quad (1)$$

where: P and T are disjoint, finite, non-empty sets of places and transitions respectively,
$Pre: P \times T \rightarrow N$ is input function,
$Post : T \times P \rightarrow N$ is output function.

There is an arc going from the p_i place to the t_j transition iff $Pre\ (p_i, t_j) > 0$. Similarly, there is an arc going from t_k place to the p_i transition iff $Post\ (t_k, p_i) > 0$.

The pre- and post-sets of transition $t \in T$ are defined respectively as $*t = \{p|Pre(p, t) > 0\}$ and $t* = \{p|Post(t, p) > 0\}$.

The practical way, in a computer analysis, of representing a net structure is pre- and post-incidence matrices of $n \times m$ dimensions; $n = |P|$, $m = |T|$. These matrices have the following elements:

$$Pre(p_i, t_j) = PRE_{ij} = \begin{cases} W\ (p_i, t_j) \text{ for } (p_i, t_j) \in {}^*t \\ 0 \text{ for } (p_i, t_j) \notin {}^*t, \end{cases} \quad (2)$$

$$Post(p_k, t_i) = POST_{ki} = \begin{cases} W\ (t_k, p_i) \text{ for } (t_k, p_i) \in t^* \\ 0 \text{ for } (t_k, p_i) \notin t^*, \end{cases} \quad (3)$$

where $W\ (p_i, t_j)$ and $W\ (t_k, p_i)$ are weights of corresponding arcs.

The dynamics of Petri net is defined by its marking (drawn as black dots). To introduce the marking, the following conditions must be fulfilled:

a) $K : P \rightarrow N^+$ each place creates capacity (capacity may be unlimited),
b) $M : P \rightarrow N^+$ is net marking, i.e. shows current number of tokens present in each place,
c) $M_0 : P \rightarrow N^+$ is the initial marking respecting the places' capacity, i.e. $M_0(p) \leq K(p)$ for all $p \in P$, N^+ – positive whole number set.

Change of state $M(p)$ proceeds during transition firing (see Fig. 2). Firing enabled transition removes a number of tokens (equal to the input arcs' weights) from each input place and adds a number of tokens (equal to the output arcs' weights) to output places.

A transition is enabled to fire if:

1) each input place has at least as many tokens as the weight of arc joining them, i.e.

$$\forall p \in {}^\bullet t : M\ (p) \geq W\ (p,\ t), \quad (4)$$

2) after firing t transition the capacity of output places cannot be exceeded, i.e.

$$\forall p \in t^\bullet : M\ (p) \geq K\ (p) - W\ (t, p). \quad (5)$$

Apart from classical nets that consist of places and transitions (P/T nets), the following concepts are applied in practice:

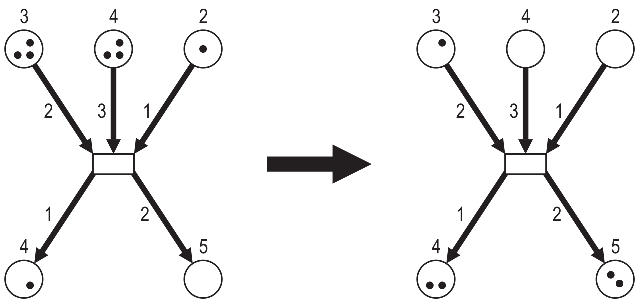

Fig. 2. The rules of transition firing

- *Timed net systems*

If Petri net is to be applied to quantitative analyses, it often needs to take account of time flow. Sequence of transitions in time can be achieved by defining a function:

$$Z : T \rightarrow R^+, \quad (6)$$

which assigns each place of a token representing a delay $z_1 = Z\ (\ t_1\ )$ (tokens in input places disappear immediately, in output places they are available with delay). The delay is either deterministic or a random variable defined by any probability distribution and can represent the process duration.

- *Inhibiting arcs*

The concept of inhibiting arcs makes it possible to control firing transitions according to actual place marking. In Fig. 3, transition t is connected with position p_2 by inhibiting arc of weight l. The transition t can be firing if there is at least one token in p_1, and the number of tokens in p_2 is lower than the weight of the inhibitor arc l.

- *Priority nets*

In the case of priority nets, a partial order in T transitions set is defined. The priority value decides about the firing sequence of enabled transitions which have at least one common place in output or input. Therefore, priority value allows allocation of limited resources.

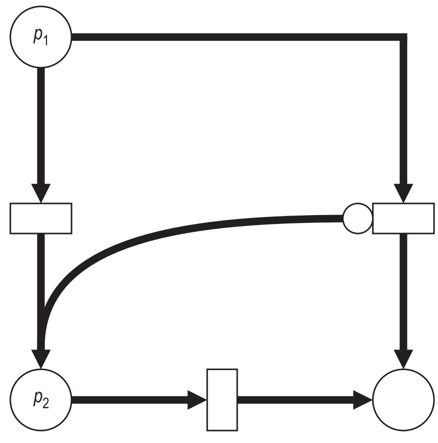

Fig. 3. Petri net with an inhibiting arc

4. Description of a computer system / support tool for scheduling

To facilitate the process of construction project scheduling, the authors created a tool based on the Petri nets concept: a computer system in Borland Delphi integrated development environment. It comprises three modules:
1. Input data.
2. Simulation programme.
3. Analysis of simulation repetition results.

4.1. Input data module

This module allows to define the following data concerning project structure and work conditions:

1. *PRE* and *POST* coincidence matrices, in which the project structure is defined and resources availability is assigned. The way of modelling the resources (such as working crew, plant and equipment) is presented in Fig. 4. This scheme is a so-called self-loop. Changes of the place marking, which represent availability of renewable resources variable in time (e.g., starting or finishing work of crew), together with corresponding dates, are recorded into simulation outcome file.

2. M_0 initial and M_K final marking complete the net description. Initial marking vector is defined as follows:

$$M_0 = [\, m_{01}, m_{02}, m_{03}, ..., m_{0m} \,], \quad (7)$$

where: m_{0i} – number of tokens in place i; $i = 1, 2, ... m$.

The only non-zero elements ($m_{0i} \neq 0$) are those that represent places standing for the project start or required resources. The project completion (through firing of transition sequence) leads to the net marking change from the initial to final value: $M_k = [m_{k1}, m_{k2}, ..., m_{0m}]$. Similarly, only these elements of the vector M_k are non-zero ($m_{ki} \neq 0$) that represent places showing the end of the project or resources release.

3. Tasks' duration: deterministic or random defined by type and parameters of distribution. In particular, the system allows the user to model the task duration as a random variable of the following distributions:

– normal distribution,
– exponential distribution,
– uniform distribution,
– user-defined distribution.

The user defines the distribution by a pair of data: duration and probability of its occurrence. Distributions are generated by reversing distribution function (see Björk and Dahlquist 1983) using generator of pseudorandom numbers defined in Object Pascal.

4. Additional conditions of firing of a given transition i, which can take the following form:

$$\begin{array}{ll} \text{(i)} & N(i) \geq k, \\ \text{(ii)} & N(i) - N(j) \geq k \ \ for\ N(l) < w, \\ \text{(iii)} & M(i) \geq k, \\ \text{(iv)} & N(i) - N(j) \leq k, \\ \text{(v)} & Z(i) = 0, \end{array} \quad (8)$$

where:
$N(i)$ – number of repetitions of fired transition i (completed repetitions of the processes run);
$M(i)$ – number of tokens in the place i;
$Z(i)$ – binary variable: $Z(i)=0$ when transition i at a given time is not being realised, $Z(i)=1$ in the opposite case;
i, k, l, w – natural numbers.

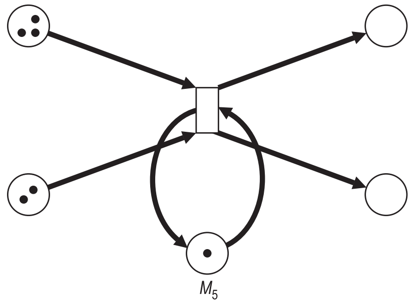

Fig. 4. The rule of construction project resources modelling

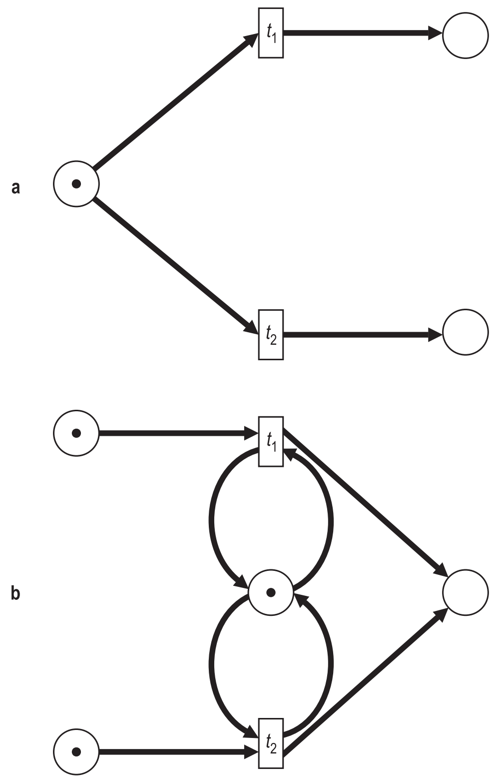

Fig. 5. Conflicts that require adjusting by means of priority rule

The idea of additional conditions is similar to that of inhibiting arcs, and facilitates the control of a construction project. It also decreases the size of the model. Condition (v) disables parallel realisation of two processes.

5. Priorities which allow the planner to solve conflicts (see Fig. 5) in case there are common input places of different transitions or resources are shared between a number of tasks.

4.2. Simulator

Fig. 6 shows the algorithm of a single replication of a simulation. The algorithm consists of the following modules:

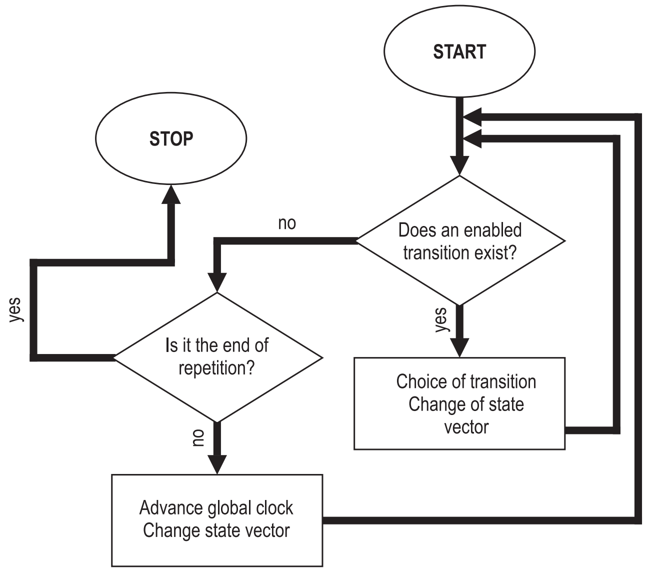

Fig. 6. Simplified diagram of simulator replication

"Start a new repetition" consists in reducing the global clock value to zero (T:=0); the marking vector takes the initial value: M:=Mo.

"Does an enabled transition exist" consists in finding all transitions that are enabled (including check of additional conditions).

"Choice of transition. Change of state vector" bases on firing transition with the highest priority among enabled ones and leads to marking vector change: $M := M - \vec{e} \times PRE$, $\vec{e} = [t_i] = [0, ..., 1, ... 0]$, where $t_i = 1$ if i equals the number of firing transitions, if not $t_i = 0$.

For the process that corresponds with firing transition, the duration τ_i is random value generated according to the distribution defined. If no enabled transition exists, **advance the global clock** is necessary:

$$T := T + \tau, \qquad (9)$$

where:

$\tau = \min_{j \in \Omega} \tau_j$,

Ω – a set of started processes (activities) that are running at the moment T.

The value of the global clock is advanced by equivalent finishing to duration of the shortest process among all that currenty proceed. It is also necessary to change the marking vector: $M := M + \vec{e} \times POST$.

If the marking vector achieves the final value, the repetition of the algorithm is terminated (**STOP**). At the end of each repetition, the simulation counter is updated ($k := k + 1$).

Simulation results, i.e.:

– project completion time,
– resource chart

are recorded in a text file and are the basis of further analyses.

The number of the replications should be stated in advance so that the relative error of mean value of the estimated parameter (e.g., project duration) at fixed confidence level does not exceed a required value (see Law and Kelton 1991).

4.3. Analysis of the results

The results of simulation replications are statistically analysed (mean and standard deviation of the project completion time and the total idle time of resources). The principle of resources' idle time calculation is shown in Fig. 7.

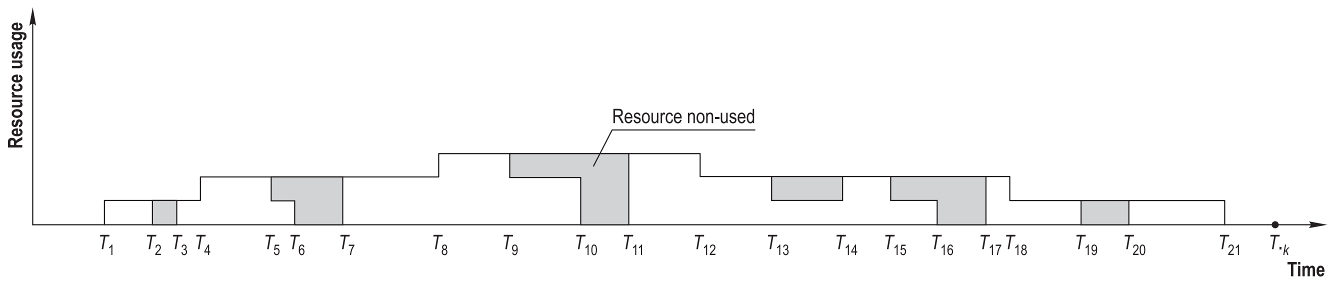

Fig. 7. Illustration of outcome file contents

5. Example

In many construction projects the same crews complete the same technological processes on the succeeding floors. Petri nets enable the planner to model repetitive operations in the form of cycle graphs. Fig. 8 presents a simplified model of a twenty-storey building project. After earthworks and foundation are completed, a series of technological processes are to be performed on each storey: cast-in-place concrete walls, assembly of steel frame and precast floor slabs, brickwork of external and partition walls, plastering, painting, flooring and windows installation. Some of the processes, i.e. earthworks, foundation work, solid floor roofing, backfilling and facade cladding, are here non-recurrent processes.

The weight of the arc joining "foundation" transition with a p_1 place is 20, hence after firing of "foundation", 20 tokens will appear in p_1 place, so repetitive processes will be performed on each floor (a storey number can be a model parameter).

The model assumes that the in-situ concrete walls of a particular floor can start, in relation to structural steel, at most two storeys in advance. Therefore, the following additional firing condition has been introduced to the transition "in-situ concrete walls":

$$N\text{ (cast-in-place concrete walls)} - N\text{ (structural steel)} \leq 1, \tag{10}$$

where: N (transition) – number of completed transitions.

A similar assumption has been made for the "precast concrete floor" and "structural steel" processes: so an additional condition has been assigned to the "structural steel" transition:

$$N\text{ (structural steel)} - N\text{ (precast concrete floor)} \leq 1. \tag{11}$$

With the "structural steel" transition, a condition of v type: Z(5) = 0 has been associated, and with the "precast concrete floor" transition – a Z(4) = 0 condition, to prevent simultaneous floor and steel structure assembly for the sake of working safety conditions.

"Roofing" can start when there are 20 tokens in p_6 place (the value of arcs joining p_6 place with "roofing" transition), so when external walls on all storeys have been completed. It has been assumed that the "external walls" of a particular storey may start no earlier than the completion of the floors of two storeys (if external walls on the last twentieth storey have not been completed yet). For this reason, an additional condition is associated with "external walls" transition:

$$N\text{ (precast concrete floor)} - N\text{ (external walls)} \geq 3 \tag{12}$$

for N (precast concrete floor) < 20.

The "backfilling" process can commence only after the first storey floor has been completed. This results in the following condition:

$$N\text{ (precast concrete floor)} \geq 1. \tag{13}$$

Distributions of particular operations' duration have been determined according to subjective probability theory, which has been described, among others, by DeGroot (1970).

5.1. Simulation research

Let us assume that project is to be managed by a general contractor, who needs to employ subcontractors to execute particular processes ("partition walls", "plastering" and "flooring') and therefore has to state the dates when the subcontractors are to start their work. The simulation is used to find the optimal dates.

The first stage of the simulation research is determining the distribution of the project duration (random variable) assuming the full availability of resources

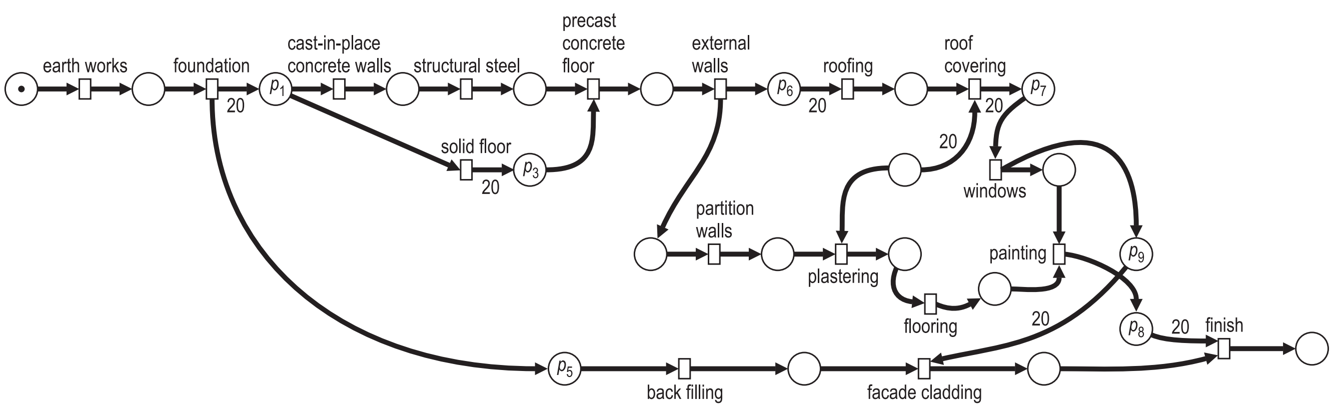

Fig. 8. Petri net model of 20-storey building construction

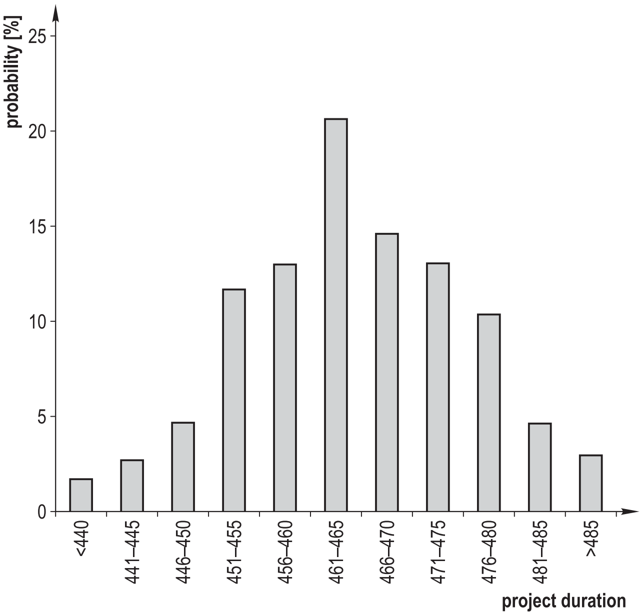

Fig. 9. Histogram of project completion time

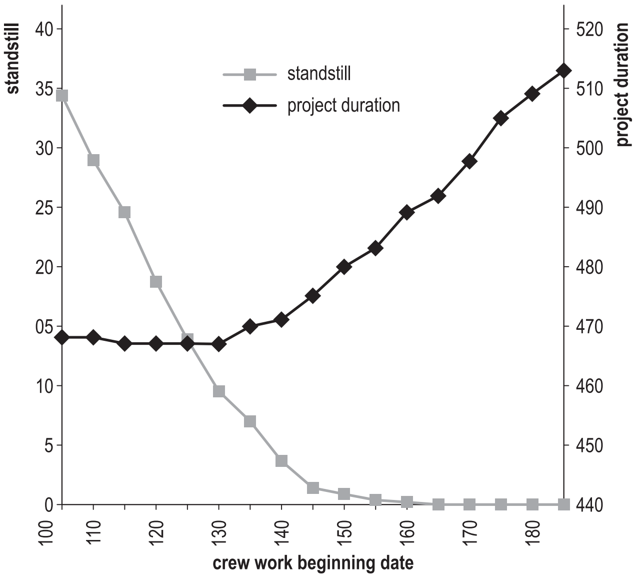

Fig. 10. Dependences between the beginning date of working crew executing external walls and their standstill days mean number and the project duration (probability equals 60%)

(crew) and without the analysis of their utilization ratio (see Fig. 9). The project duration of 468 days is planned with 60% probability and accepted by the project participants.

The second stage of the simulation research leads to determination of optimal date of employing the bricklayers to build external walls. It has been assumed that the date they start should be set so that working place is provided for them, idle time is kept to minimum and, at the same time, the date will not reduce the probability of the project finishing on time.

The dependences between the bricklayers' work start date, the mean value of their idle days and the project duration (probability kept at 60%) have been modelled and the progress of works simulated. Fig. 10 shows the results of the simulation.

In this particular case, the simulation allows the planner to find the optimal date of the bricklayers' work start – it is the 125th day of the project. The project duration does not exceed 468 days (with 60% probability) and the mean standstill of the bricklayers is 9 days.

The third stage of simulation is done to find the best date to start partition walls, and then plastering and flooring. These works may as well be executed by the same subcontractor. Therefore, the subject of the analysis is to determine the influence of the number of crews (subcontractors) on the duration of the project. Assuming that the subcontractor's resources are fully available, project duration with 60% probability does not exceed:

– 569 days in the case that one subcontractor is employed,
– 448 days when larger number of working crews will be employed.

Partition walls, and then plastering and flooring. are not critical in this case. Considering that the increase of the number of subcontractors (beyond 2) has minor effect on the project duration, and that it is difficult to predict the (front robot) for them, a decision of the planner to employ 2 subcontractors is justified.

In this third stage of simulation research, it has been assumed that working crews execute any process (partition walls, plastering, flooring) which can be initiated at the earliest because of technological dependences (for which all predecessors have been performed). Such strategy of working crews allocation allows shortening the planned project duration to 20 days.

The agreed dates of employing crews (subcontractors) to execute partition walls, plastering and flooring are established on the basis of the dependence shown in Fig. 11, found in the repetition analysis of the results.

If these working crews start their works on the 245th day, the project will be completed in 448 days with probability of 60% (the mean number of total standstill days of working crews is about 5). Their work beginning on the 275th day guarantees the project completion in the initially planned time and there are no standstills. Indirect dates (between day 245 and day 275) should be treated as comparable and efficient solutions (in Pareto terms): the project duration and standstills are minimised. To choose the final date additional preferences of decision maker should be stated.

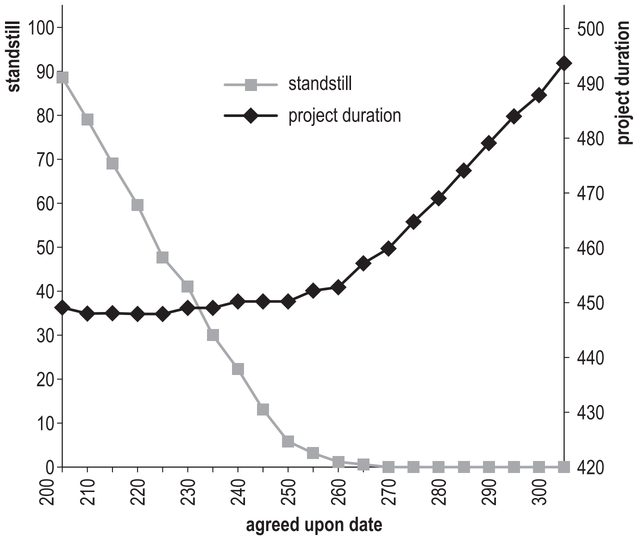

Fig. 11. Dependence between the agreed date of two working crews executing partition walls, plastering and flooring and their total standstill days mean number and project duration (with probability of 60%)

6. Conclusions

The Petri nets, in particular the PT nets presented in the paper, are a useful tool in the decision making process with regard to construction project planning. Their simple notation enables modelling the complex "real life" dependences and conditions, and at the same time facilitates programming and creation of simple structure computer systems, which allow carrying out simulations.

The Petri nets, apart from basic statistical analyses (e.g., determination of project duration with a set risk level), allow modelling resources and analysing efficiency of their use. In comparison with classical PERT approach, the method increases the range of practical application and enables scheduling construction projects, where demand for resources is determined in terms of time. The example of the Petri nets application presented in the paper substantiates the argument that the method may find use in the process of signing a contract with contractors and subcontractors in majority of construction project procurement systems.

References

Björk, Ĺ.; Dahlquist, G. 1974. *Numerical Methods.* Prentice-Hall.

Chang, J. W.; Luh, Y. P. 1997. Integration of scheduling and control in job shop, *Journal of the Chinese Institute of Engineers* 20(1): 67–76.

DeGroot, M. H. 1970. *Optimal Statistical Decisions*. Mc-Graw-Hill Book Company.

DiCesare, F.; Harhalakis, G.; Proth, J. M.; Silva, M.; Vernadat, F. B. 1993. *Practice of Petri Nets in Manufacturing*. Chapman & Hall, Londyn.

Genrich, H. J. 1991. Predicate/Transition Nets, in Jensen, K.; Rozenberg, G. (eds.). *High-level Petri-Nets. Theory and Aplication.* Spring-Verlag, Berlin, 3–43.

Halpin, D. W.; Riggs, L. S. 1992. *Planning and Analysis of Construction Operations.* John Wiley & Sons, Inc., N.Y.

Horns, A. 1989. Job shop control under influence of chaos phenomena, in *Proceedings of the IEEE International Symposium on Intelligent Control*, Albany, NY, USA - Washington, DC, USA: IEEE Computer Society Press, 227–232.

Jensen, K. 1991. Coloured Petri Nets: A high level language for system design and analysis, in Jensen, K.; Rozenberg, G. (eds.). *High-level Petri-Nets. Theory and Aplication*. Spring-Verlag, Berlin, 44–116.

Jensen, K. 1994. An introduction to the theoretical aspects of coloured Petri nets, in Bakker, J. W.; de Roever, W.-P.; Rozenberg, G. (eds.). *A Decade of Concurrency, Lecture Notes in Computer Science*. 803, Springer-Verlag.

Jensen, K. 1997. *Coloured Petri Nets*. Vol. 3. *Practical Use*. Monographs in Theoretical Computer. Springer-Verlag.

Körner, H.; Franz, V. 1989. Planung und Steuerung komplexer Bauprozesse, *Baumaschine und Bautechnik* 36(5): 247–256.

Law, A.; Kelton, W. D. 1991. *Simulation Modeling & Analysis*. McGrow-Hill.

Lin, J. T.; Lee, C. C. 1997. A Petri Net-Based integrated control and scheduling scheme for flexible manufacturing cells, *Computer Integrated Manufacturing Systems* 10(2): 109–122.

Magott, J.; Skudlarski, K. 1989. Combining generalized stochastic Petri Nets and PERT networks for the perfomance evaluation of concurrent processes, in *Proceedings of the Third International Workshop on Petri Nets and Performance Models*, Kyoyo, Japan – Los Alamitos, CA, USA: IEEE Computer Society Press, 249–256.

Reisig, W. 1982. *Petri Nets. An introduction*. Springer-Verlag, Berlin. Science, Spring-Verlag.

Sawhney, A.; Mund, A.; Marble, J. 1999. Simulation of the structural steel erection process, in *Proceedings of the 1999 Winter Simulation Conference*, 942–947.

Sawhney, A.; Abudayych, O.; Chaitavatputtiporn, T. 1999. Modeling and analysis of concrete production plant using Petri Nets, *Journal of Computing in Civil Engineering* 13(3): 178–186.

Sawhney, A. 1997. Petri Net based simulation of construction schedules, in *Proceedings of the 1997 Winter Simulation Conference*, 1111–1118.

Wakefield, R. R.; Sears, G. A. 1997. Petri Nets for simulation and modeling of construction systems, *Journal of Construction Engineering and Management* 123(2): 105–112.