NAUKI PODSTAWOWE
CZASOPISMO TECHNICZNE
TECHNICAL TRANSACTIONS
FUNDAMENTAL SCIENCES
WYDAWNICTWO POLITECHNIKI KRAKOWSKIEJ
1-NP/2010
ZESZYT 1
ROK 107
ISSUE 1
YEAR 107

LECH JAMROŻ, JERZY RASZKA*

PERFOMANCE MODEL OF CYCLIC PROCESSES

MODEL WYDAJNOŚCIOWY PROCESÓW CYKLICZNYCH

A b s t r a c t

This paper presents the use of both Petri nets and max-plus algebra formalism for modelling dynamic behaviour of system with cyclic processes. A cyclic processes system contains elements, which can operate in parallel while at the same time. They can have cooperative and competitive relationships. The research is focused on a design and control at the lowest operational level of the system. The considered processes are sequential and cyclic. As the performance evaluation measure the cycle time of the system is chosen.

Keywords: *cyclic processes, Petri nets, max-plus algebra*

S t r e s z c z e n i e

W artykule przedstawiono zastosowanie formalizmów sieci Petriego i max-plus algebry w modelowaniu systemu procesów cyklicznych. Procesy przebiegają równolegle, współpracują ze sobą lub współzawodniczą w dostępie do wspólnych zasobów. Rozważania przeprowadzono na poziomie sterowania operacyjnego. Oceny wydajności dokonano ze względu na czas cyklu pracy systemu.

Słowa kluczowe: *procesy cykliczne, sieci Petriego, max-plus algebra*

* Dr inż. Lech Jamroż, dr inż. Jerzy Raszka, Instytut Modelowania Komputerowego, Wydział Fizyki, Matematyki i Informatyki Stosowanej, Politechnika Krakowska.

1. Introduction

A considered cyclic processes system (CPS) is a clear example of systems, which exhibits parallel evolutions. In effect, a CPS contains elements, which can operate in parallel while at the same time. They can have cooperative and competitive relationships [4]. The growing demand for performance in these systems makes their design increasingly more complex. For this reason, it is important to have tools, which are adequate to their modeling. Tools of the Petri nets are especially suited to modeling considered CPS [5, 7]. Their graphic nature makes the models relatively simple and legible. The CPS needs to be analyzed from qualitative and quantitative points of view.

Qualitative analysis looks for properties like the absence of deadlocks or the presence of certain mutual exclusions in the use of shared resources [1]. Its ultimate goal is to prove the correctness of the modeled system.

Quantitative analysis looks for performance properties (e.g. cycle time, throughput) [7, 10]. This analysis concerns the evaluation of the efficiency of the modeled system.

Petri nets allow the construction of models amenable both for correctness and for efficiency analysis. Petri nets permit the modeled system to realize directly from the analyzed net. This direct implementation guarantees the permanence of the properties analyzed on the model. Net models can be used during the entire life cycle of system. The suitability of Petri nets for modeling CPS has led to their application in a wide range of fields. Examples of such systems are real time systems [11], communication networks, and parallel computing systems. Petri nets are well known as efficient tools for modelling CPS. An survey on the subject can be found in [12].

As the performance evaluation measure the cycle time of the system is chosen [7, 8, 13]. The max-plus (*alt.* MaxPlus) algebraic technique to handle systems under consideration is proposed. The max-plus algebra represents linear algebraic form of cyclic processes system. In last years, the max-plus approach has been applied to the analysis of the time behaviour of concurrent systems [2, 3, 6]. Max-plus linear systems provide natural models for which many analytical results can be applied to performance evaluation problems. For instance, problems like computing the cycle time of asynchronous digital circuits or computing the throughput of a workshop [7, 9] or of a transportation network, and performance evaluation problems for communication networks are often amenable to max-plus algebra, at least in some simplified form, see in particular [6, 2].

2. Net Representation

Petri nets, a graph-oriented formalism, allow to model and analyze systems, which comprise properties such as concurrency and synchronization.

A Petri net model of a dynamic system consists of two parts: net structure and marking. A net structure is a weighted-bipartite directed graph that represents the static part of the system. A marking is representing a distributed overall state on the structure. This separation allows one to reason on net based model at two levels – structural and behavioral. Net structure is built on two disjoint sets of objects: places and transitions, which are connected by arcs. In the graphical representation, places are drawn as circles, transitions are drawn as thin bars and arcs are drawn as arrows. Places may contain tokens,

which are drawn as dots. The vector representing in every place the number of tokens is the state of the Petri net and is referred to as its marking. This marking can be changed by the firing of the transitions. A Petri nets do not include any notion of time are aimed to model only the logical behavior of systems. The introduction of a timing specification is essential if we want to use this class of model to consider performance problem.

More formally timed Petri nets (TPN) are 5-tuples [12]: TPN = (P, T, F, M_0, τ), where $P = (p_1, p_2, ..., p_n)$, $|P| \neq 0$; $T = (t_1, t_2, ..., t_m)$, $|T| \neq 0$ is a finite disjunct set of suitable places and transitions; M_0: $P \rightarrow \mathbb{N}$ is the initial marking function which defines the initial number of tokens for every place. ($\mathbb{N} = \{0, 1, ...\}$); τ: T $\rightarrow \mathbb{R}^+$ is the firing time function, and $F \subset (P \times T) \cup (T \times P)$ is the set of arcs.

3. Algebraic Representation

To handle considered systems the authors propose max-plus algebraic technique. The max-plus algebra represents linear algebraic form of discrete systems and supplies new tools to their modelling.

Structure of max-plus algebra by definition $\mathbb{R}_{\max} = (\mathbb{R} \cup \{-\infty\}, \max, +)$ is equipped with maximization and addition operations. So, is an algebra over the real numbers with maximum and addition as the two binary operations and the identity elements are $-\infty$ and 0. It can be used appropriately to determine marking times within a given Petri net and a vector filled with marking state at the beginning.

Tools of max-plus algebra are useful to investigate properties of network. For the network, which consists of n nodes and some arcs, connecting these nodes, the time activities of nodes will be written as [1]

$$\begin{aligned} x_i(k+1) &= \max(A_{i,1} + x_1(k), A_{i,2} + x_2(k), ... A_{i,n} + x_n(k)) = \\ &= \max_j(A_{i,j} + x_j(k)), \qquad i = 1, 2, \ldots, n \end{aligned} \tag{1}$$

The max-plus algebra notation for maximization and addition operations have useful shorthand symbols $\oplus$ (pronounced ‘o-plus’) and $\otimes$ (pronounced ‘o-times’). They represent two binary operations with the max as a sum (i.e. $a \oplus b = \max(a, b)$) and the usual sum as a product (i.e. $a \otimes b = a + b$).

Using these symbols, the relation (1) becomes

$$x_i(k+1) = \bigoplus_j A_{i,j} \oplus x_j(k), \qquad i = 1, 2, \ldots, n \tag{2}$$

where:

$\bigoplus_j a_j$ – refers to maximum of elements a_j with respect to appropriate j,

$\otimes$ – refers to addition.

Coefficients of matrix $A_{i,j}$ represent sum of activity times of node j and travelling time from node j to node i of network and $x_j(k)$ is the earliest epoch at which node i becomes active for the k-th time. In vector notation (2) becomes

$$\mathbf{x}(k+1) = \mathbf{A} \otimes \mathbf{x}(k) \tag{3}$$

In practical examples, autonomous timed Petri net, which models cyclic processes (with feedback arcs), is determined by

$$\mathbf{x}(k+1) = \mathbf{A}_0 \otimes \mathbf{x}(k+1) \oplus \mathbf{A}_1 \otimes \mathbf{x}(k), \quad k = 0, 1, 2, 3, \ldots \tag{4}$$

The solution of (4) is given by [1]

$$\mathbf{x}(k+1) = \mathbf{A} \otimes \mathbf{x}(k), \quad k = 0, 1, 2, 3, \ldots \tag{5}$$

where

$$\mathbf{A} = \mathbf{A}_0^* \oplus \mathbf{A}_1$$

$$\mathbf{A}_0^* = \mathbf{I} \oplus \mathbf{A}_0^1 \oplus \mathbf{A}_0^2 \oplus \mathbf{A}_0^3 \ldots \oplus \mathbf{A}_0^1$$

and where $\mathbf{I}$ – refers identity matrix in $\mathbb{R}_{max}$; zeros on diagonal and $-\infty$ elsewhere (0 and $-\infty$ are neutral elements for max-plus algebra operations).

Matrix $\mathbf{A}_0$ and $\mathbf{A}_1$ can be determined from incidence matrix of net for its actual initial marking and times operations.

4. Computational Examples and Experimental Results

In an example three tasks {Task1, Task2, Task3} with eight operations are to be scheduled on three non-identical processors {Pr1, Pr2, Pr3}. Every one of these tasks has two or three operation and each operation may be schedule on up to three different processors. A Petri net model structure of tasks for marking vector

$$^1\mathbf{M}_0 = \{1, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0\}$$

is given in Fig. 1. Table 1 presents times of operations.

Table 1

Times of operations

	Processors		
	Pr1	Pr2	Pr3
Task1	$\tau_1 = 1$	$\tau_2 = 5$	$\tau_3 = 1$
Task2	$\tau_4 = 3$	–	$\tau_5 = 2$
Task3	$\tau_6 = 2$	$\tau_8 = 4$	$\tau_7 = 3$

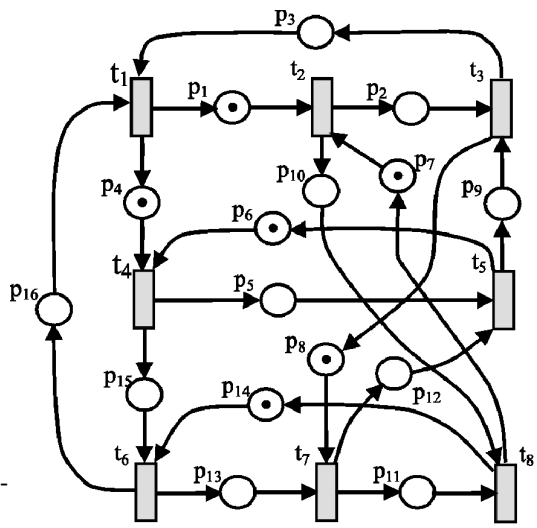

Fig. 1. Timed Petri net of the system processes
Rys. 1. Czasowa sieć Petriego modelu systemu procesów

Equation (4) for marking vector ${}^{1}\mathbf{M}_0$ is given by

$$
\begin{aligned}
x_1(k+1) &= 1 \otimes x_3(k+1) \oplus 2 \otimes x_6(k+1) \\
x_2(k+1) &= 1 \otimes x_1(k+1) \oplus 4 \otimes x_8(k+1) \\
x_3(k+1) &= 5 \otimes x_2(k) \qquad \oplus 2 \otimes x_5(k+1) \\
x_4(k+1) &= 1 \otimes x_1(k+1) \oplus 2 \otimes x_5(k+1) \\
x_5(k+1) &= 3 \otimes x_4(k) \qquad \oplus 3 \otimes x_7(k+1) \\
x_6(k+1) &= 3 \otimes x_3(k) \qquad \oplus 4 \otimes x_8(k) \\
x_7(k+1) &= 1 \otimes x_3(k) \qquad \oplus 2 \otimes x_6(k+1) \\
x_8(k+1) &= 5 \otimes x_2(k) \qquad \oplus 3 \otimes x_7(k+1)
\end{aligned}
\tag{6}
$$

From equations (6) it follows that the matrixes of coefficients $\mathbf{A}_0$ and $\mathbf{A}_1$ accept values given suitably as

$$
\mathbf{A}_0 = \begin{bmatrix}
\varepsilon & \varepsilon & 1 & \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon \\
1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 4 \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon \\
1 & \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon
\end{bmatrix}; \qquad
\mathbf{A}_1 = \begin{bmatrix}
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & 5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & 4 \\
\varepsilon & \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & 5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon
\end{bmatrix}
$$

where the symbol ε de notes value $-\infty$.

The equation (4) which has nilpotent matrix $\mathbf{A}_0$ achieves convergence for $l = 5$ i.e. $\mathbf{A}_0^5 = 0$ (all coefficients equal $-\infty$). Received result is written as

$$\mathbf{A}_0^* = \begin{bmatrix} e & \varepsilon & 1 & \varepsilon & 3 & 8 & 6 & \varepsilon \\ 1 & e & 2 & \varepsilon & 4 & 9 & 7 & 4 \\ \varepsilon & \varepsilon & e & \varepsilon & 2 & 7 & 5 & \varepsilon \\ 1 & \varepsilon & 2 & e & 4 & 9 & 7 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & e & 5 & 3 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 5 & 3 & e \end{bmatrix}$$

where the symbol e denotes the neutral element with respect to addition, it assumes the numerical value 0.

The final transient matrix $\mathbf{A}$ in equation (5) is written as

$$\mathbf{A} = \begin{bmatrix} \varepsilon & 6 & 7 & 11 & \varepsilon & \varepsilon & \varepsilon & 12 \\ \varepsilon & 9 & 8 & 12 & \varepsilon & \varepsilon & \varepsilon & 13 \\ \varepsilon & 5 & 6 & 10 & \varepsilon & \varepsilon & \varepsilon & 11 \\ \varepsilon & 7 & 8 & 12 & \varepsilon & \varepsilon & \varepsilon & 13 \\ \varepsilon & \varepsilon & 4 & 8 & \varepsilon & \varepsilon & \varepsilon & 9 \\ \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & 4 \\ \varepsilon & \varepsilon & 1 & 5 & \varepsilon & \varepsilon & \varepsilon & 6 \\ \varepsilon & 5 & 4 & 8 & \varepsilon & \varepsilon & \varepsilon & 9 \end{bmatrix}$$

The results of iterative calculation of $\mathbf{x}(k)$ for $k = 0, 1, 2 \ldots$ according to the dependencies (5) becomes

$$\mathbf{x}(0) = \begin{bmatrix} \varepsilon \\ 0 \\ \varepsilon \\ 0 \\ \varepsilon \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{bmatrix}; \quad \mathbf{x}(1) = \begin{bmatrix} 11 \\ 12 \\ 10 \\ 12 \\ 8 \\ 3 \\ 5 \\ 8 \end{bmatrix}; \quad \mathbf{x}(2) = \begin{bmatrix} 23 \\ 24 \\ 22 \\ 24 \\ 20 \\ 15 \\ 17 \\ 20 \end{bmatrix}; \quad \mathbf{x}(3) = \begin{bmatrix} 35 \\ 36 \\ 34 \\ 36 \\ 32 \\ 27 \\ 29 \\ 32 \end{bmatrix};$$

These values in following cycles k represent times of start activity suitable transitions in considered Petri net. These results find confirmation in passed simulation using other applications [10]. Figure 2 shows suitably firing sequence of transition for the vector ${}^1\mathbf{M}_0$.

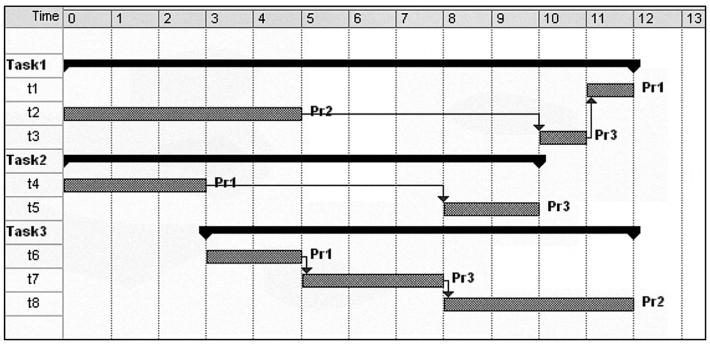

Fig. 2. Sequence firing of transitions
Rys. 2. Diagram palenia się tranzycji

5. Analysis of Petri Net Properties

The max-plus algebra is used for examined selected properties: safety, liveness, deadlock freeness and periodicity. Some of them are presented for different modification in starting location of marking. The results of iterative calculation (5) for initial marking vector

$$^{2}\mathbf{M}_0 = \{0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0\}$$

becomes

$$\mathbf{A}_0^{100} = \begin{bmatrix} \varepsilon & \varepsilon & 232 & \varepsilon & \varepsilon & 233 & \varepsilon & \varepsilon \\ 232 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 321 & \varepsilon \\ \varepsilon & 236 & \varepsilon & \varepsilon & 233 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 231 & \varepsilon \\ 232 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 199 & \varepsilon & \varepsilon & 200 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 233 & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}$$

and

$$\mathbf{x}(0) = \begin{bmatrix} \varepsilon \\ 0 \\ \varepsilon \\ 0 \\ \varepsilon \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{bmatrix}; \quad \mathbf{x}(1) = \begin{bmatrix} 230 \\ 231 \\ 235 \\ 1 \\ 8 \\ 3 \\ 5 \\ 235 \end{bmatrix}; \quad \mathbf{x}(2) = \begin{bmatrix} 469 \\ 470 \\ 471 \\ 470 \\ 244 \\ 239 \\ 241 \\ 471 \end{bmatrix}; \quad \mathbf{x}(3) = \begin{bmatrix} 705 \\ 706 \\ 707 \\ 706 \\ 480 \\ 475 \\ 477 \\ 707 \end{bmatrix};$$

Figure 3 shows suitably firing sequence of transition for the vector ${}^{2}\mathbf{M}_0$.

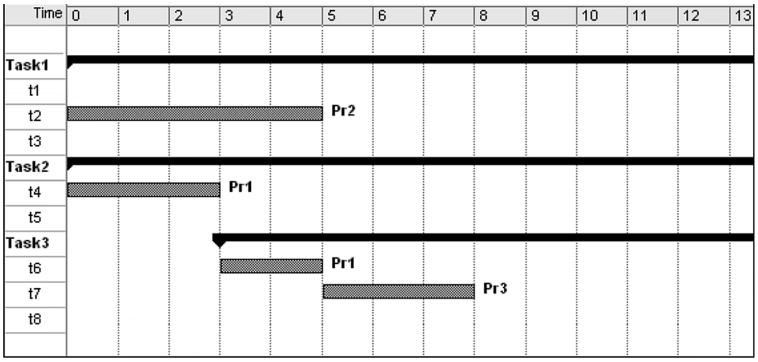

Fig. 3. Firing of transitions for vector ${}^{2}\mathbf{M}_0$
Rys. 3. Diagram palenia się tranzycji dla wektora ${}^{2}\mathbf{M}_0$

It is not possible to obtain convergence in the iterative $\mathbf{A}_0^*$ calculation as a deadlock has occurred within system.

6. Concluding Remarks

The obtained results confirm that the method presented here is useful for some classes cyclic processes systems, which are modelling by Petri nets. An algebraic description of processes permits to analyse some property of systems. This method is suitable for analysing the effect of changes of the initial state of a system. In particular, it is helpful in determining a periodicity, a time of a cycle and to detect the initial state of the system, which carries to deadlock.

References

[1] Bacceli F.L., Cohen G., Olsder G.J., Quadrat J.P., *Synchronization and Linearity. An Algebra for Discrete Event Systems*, John Wiley & Sons Ltd., London 1992.

[2] Bacceli F., Hong D., *TCP is max-plus linear and what it tells us on its throughput*, Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Communication, 2000, 219-230.

[3] Benveniste A., Gaubert B.A., Jard C., *Monotone rational series and max--plus algebraic models of realtime systems*, Proc. of the FourthWorkshop on Discrete Event Systems (WODES98), IEE, Cagliari, Italy 1998.

[4] Cassandras Ch.G., Lafortune S., *Introduction to Discrete Event Systems*, Springer 2008, Kluwer Academic Publishers, 2007.

[5] Di Cesare F. et al., *Practice of Petri nets in manufacturing*, Chapman & Hall, London 1993.

[6] Elmahi I., Grunder O., Elmoudni A., *A max plus algebra approach for modeling and control of lots delivery*, Industrial Technology, Vol. 2, IEEE ICIT '04, 8–10 Dec. 2004, 926-931.

[7] Hillion H., Proth J.M., *Performance evaluation of job shop systems using timed event graphs*, IEEE Trans. Automat. Control, Vol. 34, April 1989.

[8] Jamroż L., *Zastosowanie metod ewolucyjnych w optymalizacji cyklicznych procesów dyskretnych*, Automatyka, t. 5, z. 1, Wyd. AGH, Kraków 2001.

[9] Jamroż L., *A method for work-in-process reduction in Flexible Manufacturing System*, Proceedings of 10th International Symposium on System Modelling Control, Zakopane 2001, 221-226.

[10] Jamroż L., Raszka J., *Simulation method for the performance evaluation of system of discrete cyclic processes*, Proceedings of the 16th IASTED International Conference on Modelling, Identification and Control, Innsbruck, Austria Feb. 17–19th 1997, 190-193.

[11] Kwiecień A., Gaj P. (red.), *Systemy informatyczne z ograniczeniami czasowymi*, WKiŁ, Warszawa 2006.

[12] Murata T., *Petri nets: properties, analysis and applications*, Proceedings of the IEEE, Vol. 77, No. 4, 1989, 541-580.

[13] Villani E., Paulo E., Valette R., *Modelling and Analysis of Hybrid Supervisory Systems*, Series: Advances in Industrial Control, Springer, 2008.