**TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE**
FUNDAMENTAL SCIENCES | NAUKI PODSTAWOWE
1-NP/2016

JERZY RASZKA*, LECH JAMROŻ*

# ANALYSIS AND DESIGN OF CONTROL DATA PROCESSING AS DISCRETE EVENT SYSTEMS

# ANALIZA I PROJEKTOWANIE STEROWANIA PRZETWARZANIEM DANYCH JAKO SYSTEMU ZDARZEŃ DYSKRETNYCH

A b s t r a c t

The following article presents the application of the Max-Plus Linear System (MPLS) method in the synthesis of different IT process control structures, taking place in the systems belonging to the class of Discrete Event Systems (DES). Modelling and analysis of these systems are based on the state equations, expressed in MPLS categories and classical principles of control theory, adapted for them. MPLS is based on max-plus algebra formalism and is supported with graphical representation in the form of Timed Event Graph (TEG), which is the special case of Timed Petri Nets (TPN). The article contains an overview of the theoretical work on discrete control processes, with a particular focus on the synthesis of control signals in the open control to obtain the desired output system. A practical example has been used for distributed computational process and data transmission for the system controlling selected technological process. Numerical results are presented.

*Keywords: max-plus linear system, timed event graph, event system, control*

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

W artykule przedstawiono zastosowanie metod opartych na max-plus liniowym systemie (MPLS) w syntezie struktur sterujących różnych procesów IT, zachodzących w systemach należących do klasy systemów zdarzeń dyskretnych. Modelowanie i analiza tych systemów bazuje na równaniach stanu, wyrażonych w kategoriach MPLS i przystosowanych do nich, klasycznych zasadach teorii sterowania. MPLS opiera się na formalizmie max-plus algebry i posiada reprezentację graficzną w postaci czasowego grafu zdarzeń, który jest szczególnym przypadkiem czasowych sieci Petriego. Artykuł zawiera przegląd prac teoretycznych dotyczących sterowania procesami zdarzeń dyskretnych ze szczególnym przedstawieniem syntezy sygnałów w otwartym układzie sterowania mających na celu wymuszenie zadanej odpowiedzi. Praktyczny przykład zastosowano dla rozproszonego procesu obliczeniowego z przesyłaniem danych do systemu sterowania wybranego procesu technologicznego oraz przedstawiono wyniki obliczeń numerycznych.

*Słowa kluczowe: max-plus liniowy system, czasowy graf zdarzeń, system zdarzeń, sterowanie*

**DOI: 10.4467/2353737XCT.16.151.5762**

* Jerzy Raszka (jraszka@riad.pk.edu.pl), dr inż. Lech Jamroż, Institute of Computer Science, Cracow University of Technology.

## 1. Introduction

The increasing complexity of information processes in distributed computer systems and microprocessor systems increases the probability of faults and disturbances. Hence, there is the need to include them in the design process. These processes were treated as occurring in class of discrete event systems (DES).

The DES class is very wide, covering manufacturing systems (including flexible and assembly lines) [5, 6], road, railway and air transport systems [3, 19], as well as computer networks [15]. In addition, one can also qualify processes in the field of Human System Interaction, e.g. resource and task management, as well as process control technologies. The variety of DES systems leads to different models. The MPLS model has been adopted in this article, based on the algebra theory (max, +). One of the first scientists, who described this theory was R.A. Cuninghame-Green [12] and then it was developed by the INRIA team [7]. The authors of the articles showed that the behaviour of certain DES systems can be described using linear equations. They have also pointed at many cases of analogies to the problems in the systems theory, automation and control. Joint researches led to the publication of collective work [2]. In the following article, based on the bibliographic data, the theoretical basis of modelling and control in the DES systems are discussed, supplemented with the authors' results from the scope of this problem.

In the area of research, in particular should be mentioned:
- Optimal control in the open loop. Structure described by the Cohen [9], in which a well-known system model and time sequence of output signals are assumed, while the optimal trajectory of the input signals is calculated,
- Preliminary compensator [14, 18]. The time sequence of output signals is not known, but the reference model, which imposes the behaviour of the output relative to the input is assumed.
- Corrector with the feedback. In the control structure the system output is modified by the correctors in the feedback [10]. They converge the behaviour of the whole system to the behaviour set in the reference model.
- R, S, T type correction. Strategy based on the introduction of three correctors into the structure, has been inspired by the Åström [1]. This leads to the better results than those obtained with the single corrector [20].
- Control in the presence of disturbances [16]. System is exposed to the acting of uncontrolled inputs. To reduce their impact, the control in the closed-loop is taken into account.
- Robust control [17]. It is assumed that the system parameters are random, but are in the specific range of values. Synthesis of control is based on the feedback.

The main part of the following article concerns the design of the open control structures, including control under conditions of uncertainty in the data transmission aspect.

This article is organized as follows. Section 2 introduces the maxplus algebra theory and MPLS modelling. In section 3, based on the literature review, the authors' results related to the selected problems of processes control in the DES systems have been presented. Also some problems in the disturbances conditions, damages and uncertainty have been discussed. In section 4 there is the general theory related with the open control. In section 5 control system synthesis has been described and the practical results for the computational processes and data transmission in model of IT systems have been presented.

## 2. Mathematical fundamentals

This section contains selected basic concepts of max-plus algebra providing the basis to formulate a model MPLS. They are widely discussed in the Chapters 3 and 4 of publication [2]. Max-plus algebra formalism is based largely on the lattice theory and partially ordered sets, and residuation theory. In turn, the theoretical basis, allowing representation of MPLS in the categories of the state equations system is presented in Chapters 5 and 6 of [2].

### 2.1. Max plus algebra [25]

In recent years, the concept of a max-plus-linear system (MPLS) has been increasingly frequently used in the literature. It is based on a mathematical formalism, namely max-plus algebra. The basic operations of max-plus algebra are maximization and addition, which will be represented, respectively, by $\oplus$ and $\otimes : x \oplus y = \max(x, y)$ and $x \otimes y = x + y$ for $x, y \in R_\varepsilon$, $R_\varepsilon \stackrel{def}{=} R \cup \{-\infty\}$

The reason for using these symbols is that there is a remarkable analogy between $\oplus$ and conventional addition, and between $\otimes$ and conventional multiplication: many concepts and properties from linear algebra (such as the Cayley-Hamilton theorem, eigenvectors and eigenvalues, Cramer's rule) can be translated to max-plus algebra by replacing + with $\oplus$ and $\times$ with $\otimes$. Hence we also call $\oplus$ the max-plus-algebraic addition, and $\otimes$ the max--plus-algebraic multiplication. Note, however, that a major difference between conventional algebra and max-plus algebra is that, in general, there are no inverse elements with respect to $\oplus$ in $R_\varepsilon$. The zero element for $\oplus$ is $\varepsilon \stackrel{def}{=} -\infty$ and we have $a \oplus \varepsilon = a = \varepsilon + a$ for all $a \in R_\varepsilon$. The structure $(R_\varepsilon, \oplus, \otimes)$ is referred to as max-plus algebra. Let $r \in R$. The $r^{\text{th}}$ max-plus-algebraic power of $x \in R$ is denoted by $x^{\otimes r}$ and corresponds to $rx$ in conventional algebra. If $x \in R_\varepsilon$, then $x^{\otimes 0} = 0$ and the inverse element of $x$ w.r.t. $\otimes$ is $x^{\otimes -1} = -x$. There is no inverse element for $\varepsilon$ since $\varepsilon$ is absorbing for $\otimes$. If $r > 0$, then $\varepsilon^{\otimes r} = \varepsilon$, and if $r < 0$, then $\varepsilon^{\otimes r}$ is not defined. In this paper, we have $\varepsilon^{\otimes 0} = 0$ by definition.

The implicit equation $x = a \otimes x \oplus b$ determines $a = a^* \otimes b$ where the Kleene star operator:

$$a^* = \bigoplus_{i=0}^{\infty} a^i$$

The rules for the order of evaluation of max-plus algebraic operators correspond to those of conventional algebra. So the max-plus-algebraic power has the highest priority, and max--plus-algebraic multiplication has a higher priority than max-plus-algebraic addition.

The basic max-plus-algebraic operations are extended to matrices as follows.

If $\mathbf{A}, \mathbf{B} \in R_\varepsilon^{m \times n}$ and $\mathbf{C} \in R_\varepsilon^{m \times p}$, then:

$$(\mathbf{A} \oplus \mathbf{B})_{ij} = a_{ij} \oplus b_{ij} = \max(a_{ij}, b_{ij})$$

$$(\mathbf{A} \otimes \mathbf{C})_{ij} = \bigoplus_{k=1}^{n} a_{ik} \otimes c_{kj} = \max_{k=1\ldots n} (a_{ik} + c_{ki})$$

for all *i, j*. Note the analogy with the definitions of the matrix sum and the product in conventional linear algebra.

The matrix $\boldsymbol{\varepsilon}_{m\times n}$ is the $m\times n$ max-plus-algebraic zero matrix: $(\boldsymbol{\varepsilon}_{m\times n})_{i,j} = \varepsilon$ for all $i, j$; and the matrix $\mathbf{E}_n$ is the $n \times n$ max-plus-algebraic identity matrix: $(\mathbf{E}_n)_{ii} = 0$ for all $i$ and $\mathbf{E}(\mathbf{E}_n)_{ij} = \varepsilon$ $i, j$ with $i \neq j$. If the size of the max-plus-algebraic identity matrix or the max--plus-algebraic zero matrix is not specified, it should be clear from the context. The max-plus--algebraic matrix power of $\mathbf{A} \in R_{\varepsilon}^{n\times n}$ is defined as follows: $\mathbf{A}^{\otimes 0} = \mathbf{E}_n$ and $\mathbf{A}^{\otimes k} = \mathbf{A} \otimes \mathbf{A}^{\otimes(k-1)}$ for $k = 1,2, \ldots$

The Kleene star operator can also be applied to matrices:

$$\mathbf{A}^* = \bigoplus_{i=0}^{\infty} \mathbf{A}^i \quad \text{with} \quad \mathbf{A}^{i+1} = \mathbf{A} \otimes \mathbf{A}^i \text{ and } \mathbf{A}^0 = \mathbf{E} \tag{1}$$

where:

**E** – the identity matrix.

Equation (1), which has nilpotent matrix, achieves convergence (all coefficients equal ε).

### 2.2. Model of the system

In article [5] Cohen showed that the nonlinear dynamic systems, whose structure and behaviour is based on the timed event graph (TEG) may be described using the linear equation system The example of a TEG with the determined holding time of 2 units in place P1 is given in Fig. 1 [4].

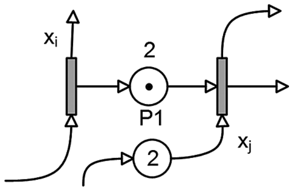


Fig. 1. Graphical representation of a TEG

State-space descriptions in the max-plus algebra for a certain class of discrete-event-systems become linear representations which are similar to state-space equations in the traditional model control theory. Generally speaking, for any TEG system, one obtains the following kind of equations as an MPLS [8]:

$$\mathbf{x}(k) = \bigoplus_{i=0}^{M} \mathbf{A}_i \ \mathbf{x}(k-i) \oplus \mathbf{B}_i \, \mathbf{u}(k-i) \tag{2.1}$$

$$\mathbf{y}(k) = \bigoplus_{i=0}^{M} \mathbf{C}_i \ \mathbf{x}(k-i) \tag{2.2}$$

where $\mathbf{x}$, $\mathbf{u}$, and $\mathbf{y}$ are vectors of dimensions equal to the numbers of internal, input and output transitions, respectively. $\mathbf{A}_i$, $\mathbf{B}_i$, and $\mathbf{C}_i$ are matrices of the appropriate dimensions with entries in the max-plus algebra, and $M$ is the maximal number of tokens in the initial marking. The variables of (2) are time instances and the represented events occur at $k$-times. The coefficients of matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ represent parameters associated with the places located between these transitions. The classical theory of the continuous and discrete systems in the time domain, revolutionized the integral transforms (e.g. Laplace, Fourier, Z-transform). Similar transformations have become useful in the theory of discrete processes. Each transition in the TEG model can be assigned to the appropriate of both, input or output vector's components, as well as to internal state.

In the article [9] is derived model, of the system is represented by 2-dimentional $(\gamma, \delta)$ – transform noted as $\mathbf{M}_{\min}^{\max}[[\gamma,\delta]]$ a set of formal power series for two variables $\gamma$ and $\delta$. A finite series of $\mathbf{M}_{\min}^{\max}[[\gamma,\delta]]$ is a polynomial and is used to code a set of information concerning the transition of a TEG. The monomial $\gamma^k\delta^t$ may be interpreted as the $k$-th event occurring at least at time $t$.

Using transform by $\mathbf{M}_{\min}^{\max}[[\gamma,\delta]]$ the TEG system (2) has implicit form as

$$\mathbf{x} = \mathbf{A}\mathbf{x} \oplus \mathbf{B}\mathbf{u} \tag{3.1}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} \tag{3.2}$$

where $\mathbf{A} \in \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_{n\times n}$, $\mathbf{B} \in \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_{n\times p}$, $\mathbf{C} \in \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_{m\times n}$,

System equation (3) by Kleene star (1) transform, gives the explicit form as

$$\mathbf{x} = \mathbf{A}^*\mathbf{B}\mathbf{u} \tag{4.1}$$

$$\mathbf{y} = \mathbf{H}\mathbf{u} \tag{4.2}$$

where $\mathbf{H} = \mathbf{C}\mathbf{A}^*\mathbf{B} \in \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_{p\times m}$ is input/output transfer matrix relation

System equation (3) and matrix $\mathbf{H}$ is modelled as block schema (Fig. 2).

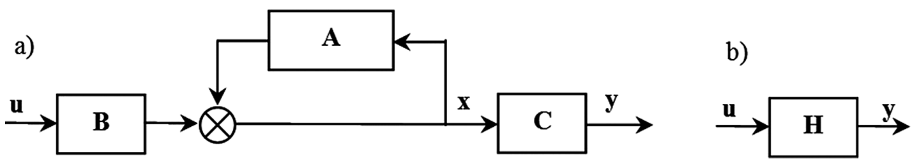


Fig. 2. Block schema of the system (a), and its substitute (b)

## 3. Control systems

### 3.1. Open control

First results concerning the open control, obtained using the (max,+) algebra are included in the Cohen [9] and Menguy [21] articles. Control was proposed for developing set *a priori* output trajectory, specified as open control. This control plays a key role in event planning [21] and scheduling of tasks. For example these tasks can be executed by the some processes in a distributed computing micro-processors system. This issue will be presented in detail with example in sections 5 and 6.

### 3.2. Feedback control from the output

Controlled structure consists with the corrector between the system's output **y** and its input **v**. Output signals of events are modified by the corrector and put together with the input events. This problem is described in details in the work of B. Cottenceau [10] and in the articles [15]. Structure of control with output feedback as the block diagram is explained in the Fig. 3.

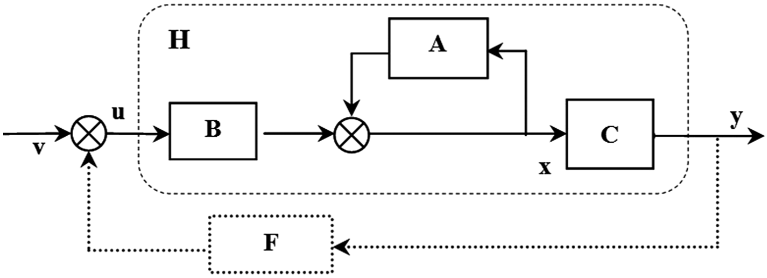


Fig. 3. Control with of the feedback from the output

For this system $\mathbf{u} = \mathbf{F}\mathbf{y} \oplus \mathbf{v}$

$$\mathbf{y} = \mathbf{H}(\mathbf{F}\mathbf{y} \oplus \mathbf{v}) = \mathbf{H}\mathbf{F}\mathbf{y} \oplus \mathbf{H}\mathbf{v}$$

and using (1) $\mathbf{y} = (\mathbf{H}\mathbf{F})^{*}\mathbf{H}\mathbf{v} = \mathbf{G_F}\mathbf{v}$

where
$$\mathbf{G_F} = (\mathbf{H}\mathbf{F})^{*}\mathbf{H} \leqslant \mathbf{G_z} \tag{5}$$

Expression (5) may be used to find **F** as the best control for applied desired characteristics and may be at least as fast as the reference $\mathbf{G_z}$.

### 3.3. Feedback control from the state

Structure of control with state feedback is explained in Fig. 4. In this case change of control structure consists of the corrector between the system's output and input. System is being controlled using by signal of state system's events, and changing by the corrector **F** analogically as in previous subsection 3.2 modified input the system. This problem is described in details in the work of B. Cottenceau [10] and in the article [15].

For this system $\mathbf{u} = \mathbf{F}\mathbf{x} \oplus \mathbf{v}$

and $\mathbf{x} = \mathbf{A}\mathbf{x} \oplus \mathbf{B}\mathbf{u} = \mathbf{A}\mathbf{x} \oplus \mathbf{B}\mathbf{K}\mathbf{x} \oplus \mathbf{B}\mathbf{v}$

$$\mathbf{x} = (\mathbf{A} \oplus \mathbf{B}\mathbf{F})\mathbf{x} \oplus \mathbf{B}\mathbf{v}$$

and solve using (1) $\mathbf{x} = (\mathbf{A} \oplus \mathbf{B}\mathbf{F})^{*}\mathbf{B}\mathbf{v}$

Output $$\mathbf{y} = \mathbf{C}(\mathbf{A} \oplus \mathbf{BF})^* \mathbf{Bv} = \mathbf{G_F v} \tag{6}$$

where $$\mathbf{G_F} = \mathbf{CA}^*(\mathbf{A}^*\mathbf{BF})^* \mathbf{A}^*\mathbf{B} = \mathbf{CA}^*\mathbf{B}(\mathbf{FA}^*\mathbf{B})^* \tag{7}$$

$$\mathbf{G_y} \leqslant \mathbf{G_z} \tag{8}$$

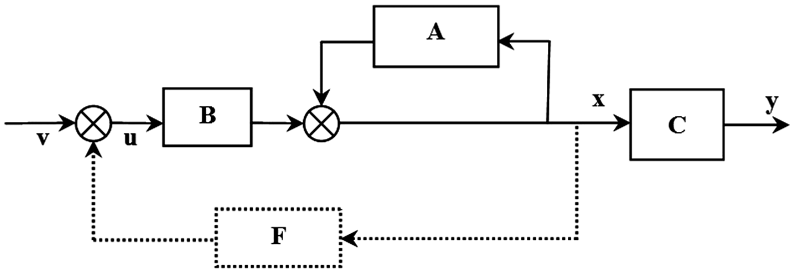


Fig. 4. Feedback control from the state

Expressions (7) may be used to find **F** as the best control for applied desired characteristics and may be at least as fast as the reference $\mathbf{G_z}$ (8).

### 3.4. Control with the observer

The availability of state of the system in the previous point, is an important condition but not always possible to fulfil. There is, however, based on a known model of the system can calculate the analytical condition which is reconstructed state. On the basis of the analogue, a conventional approach Fig. 5 shows the structure of an observer [26].

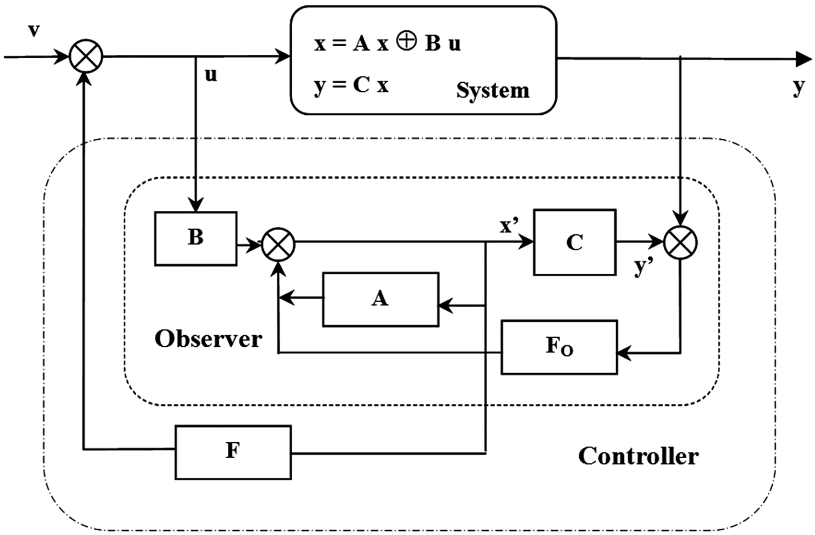


Fig. 5. Structure of control with the observer

Assuming the existence of matrices **A**, **B** and **C** the real system equation (3) in explicit form is

$$\mathbf{x} = \mathbf{A}^{*}\mathbf{B}\mathbf{u}$$

and the equation of the observer is

$$\mathbf{x}' = \mathbf{A}^{*}\mathbf{B}\mathbf{u} \oplus \mathbf{Fo}(\mathbf{y} \otimes \mathbf{y}') \tag{9}$$

Our goal is to calculate the matrix $\mathbf{F_O}$ to ensure that the estimated output $\mathbf{y}'$ is less than or equal to the measured output $\mathbf{y}$.

Typically, an observer is used to estimate the conditions necessary for feedback from the state and it is possible now to use control as in subsection 3.3 with equation (6)

$$\mathbf{y} = \mathbf{C}(\mathbf{A} \oplus \mathbf{BF})^{*}\mathbf{B}\mathbf{v} \tag{10}$$

Now expressions (9) and (10) may be used to find $\mathbf{F}$ and $\mathbf{F_O}$ as the best control for applied desired characteristics.

## 4. Data processing with control

Let us consider a data process that allows event-driven applications to take advantage of multiprocessors by running the code for event handlers in parallel. To achieve high performance, servers must overlap computation with the I/O. Programs typically achieve this overlap by using threads or events. Threaded programs usually process every request in a separate thread; while one thread block is waiting for the I/O, another thread can run. Event-based programs are structured as a collection of call-back functions which are called by the main loop when I/O events occur. Threads provide an intuitive programming model, but require coordinating the access of different threads to the shared state, even on a uniprocessor. Event-based programs execute call-backs sequentially so the programmer need not worry about concurrency control; however, event-based programs have so far been unable to make good use of multiprocessors. Much of the effort required to make existing event-driven programs take advantage of multiprocessors is in specifying which events can be handled in parallel.

This article presents a simple problem of designing the control of a system in which the cost is chosen so that it provides a trade-off between minimizing the delays of the end time of computational process operations (the real time to complete all the tasks in a cyclic computational process, times of final results of one cycle) and the periodicity of the desired output (the time desired or needed) to complete the process.

This problem was presented with no disturbances [22] and it was solved in max-plus algebraic functions as dater equation. Now we introduce disturbances and this problem is modelled in the 2-dimensional $\mathbf{M}_{\min}^{\max}[[\gamma, \delta]]$ domain.

Simple data processing consists of several tasks linked by the wait for I/O data (Fig. 6). To illustrate our approach, let us consider a process that consists of some tasks: $T_{0i}$ which runs on microprocessors: $\mu P_{0i}$, for $i$ =1, ..., $n$. Each of these tasks is executed on a dedicated microprocessor. In this process, the digital information flows as input/output processing data

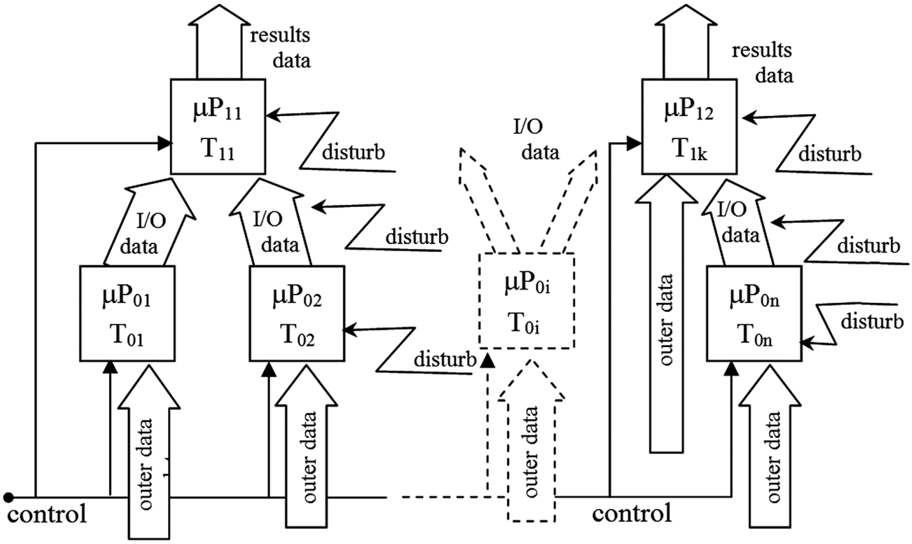


Fig. 6. The structure of the disturbing processes

and a control signal. Input data (i.e. from outer system) is processed as the first task on the $\mu P_{01}$ and its output data has to be saved to memory while it waits to be processed. The other microprocessors operate in the same way, but their input data simultaneously constitutes the output result from the previous microprocessor and /or may need extra outer data too.

## 5. Control in open structure

In this section, the specified project is considered to obtain open control problem. In the context of data processing, this problem is to give final results and at the same time minimizing the size of the memory. Specifically, the control problem in open loop resolved as follows. It is system (a TEG with p inputs and q outputs) whose transfer matrix is known to transfer $\mathbf{H} = \mathbf{CA}^*\mathbf{B} \in \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_{p\times m}$. It is desired, using inputs $\mathbf{u} \in \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_p$ to ensure that the system outputs follow the best trajectory determined by $\mathbf{z} \in \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_p$.

In [8], it is shown that this problem has an optimal solution, that there is a greater input control $\mathbf{u_{opt}} \in \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_p$ such that the output resulting from that input ($\mathbf{y_{opt}} = \mathbf{Hu_{opt}}$) is less than or equal to the desired output $\mathbf{z}$. The $\mathbf{u_{opt}}$ order is optimal from the point of view the just-in-time criteria ($\mathbf{y_{opt}}$ the output is just-in-time). Here we implement restrictions.

- Input reference can be updated. For example, in the context of data processing the final results may lead to modifications of outer processes.
- Deadlines for the firing of some of the input transition can't be modified, which may provide input data to the actual processes.

Formally, transformation of $L_H : \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_p \Rightarrow \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_p,\ u \Rightarrow H \otimes u,$ defines optimal control.

$$\{\mathbf{u} \in \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_p \,|\, L_H(u_{\text{opt}}) \leqslant z\}.$$

More specifically this is the upper limit (marked $u_{\text{opt}}$), which gives you the greatest control satisfying the condition of $L_H(u_{\text{opt}}) \leqslant z.$ We can already see that this set is not empty since $u = \varepsilon$ is the solution, that is $L_H(\varepsilon) \leqslant z$ and it is inversion problem which the theory residuation solves this problem directly.

The optimal command $u_{\text{opt}}$ exists and is given by:

$$\mathbf{u}_{\mathbf{opt}} = \{\mathbf{u} \in \mathbf{M}_{\min}^{\max}[[\gamma,\delta]]_p \,|\, L_H(u) \leqslant z\} L_H(z) = H \setminus z \tag{11}$$

The optimal control for TEG corresponds to the order by entering the markers to the system as late as possible.

## 6. Example

In order to accomplish achieve the results, we'll look at an example of the system processes. Consider the TEG model in Fig. 7. As mentioned in section 4 this model can represent i.e. a tasks of a process in a distributed computing system constructed of some micro-processors P and memory units T. In this example data results from P1, P2 and P5 is buffered in T5, T6, T7 and then there are processed by P3 and P4. Note that processors P1, … P5 have different cycle times: i.e P1 can handle a task every 2 units while P2 every 4 units of time etc. For this system, according to $\mathbf{M}_{\min}^{\max}[[\gamma,\delta]]$ representation (3,4) we have

$$\mathbf{A} = \begin{bmatrix} \varepsilon & \gamma & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \gamma & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \delta^4 & \varepsilon & \varepsilon & \gamma & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \delta^5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta^2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \delta^6 & \varepsilon & \varepsilon & \varepsilon & \gamma & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta^3 & \varepsilon \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \delta^1 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \delta^1 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \delta^2 \\ \varepsilon & \varepsilon & \varepsilon \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & \varepsilon \end{bmatrix}$$

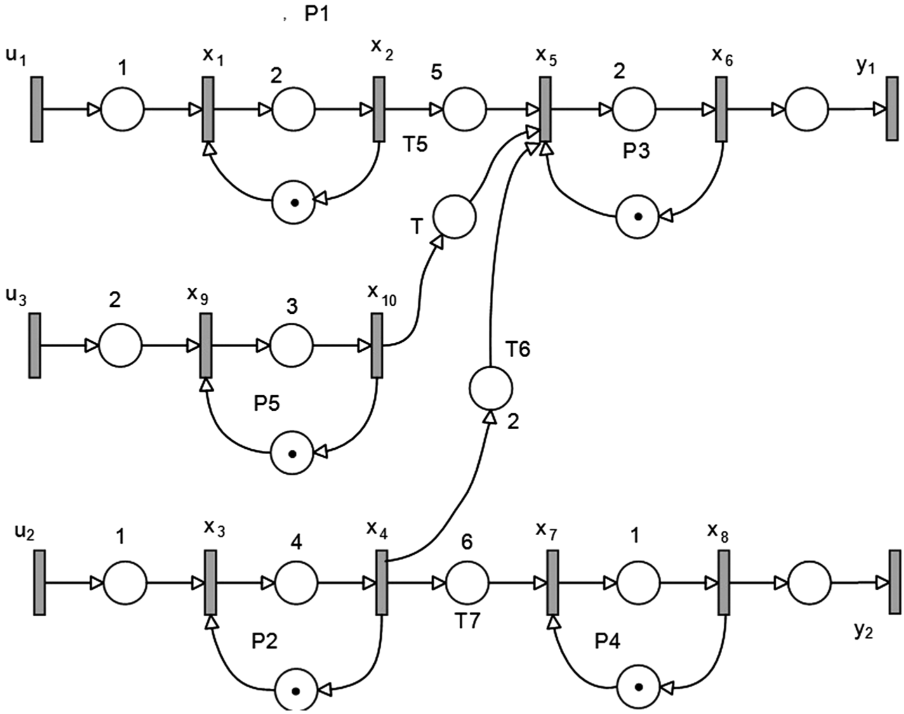


Fig. 7. TEG of the system processes

According to (4) and, we can rewrite system transfer

$$\mathbf{H} = \mathbf{C}\mathbf{A}^{*}\mathbf{B} = \begin{bmatrix} \delta^{10}(\gamma\delta^{2})^{*} & \varepsilon & \varepsilon \\ \varepsilon & \delta^{12}(\gamma\delta^{4})^{*} & \varepsilon \end{bmatrix}$$

We may to determine a desired output i.e.

$$\mathbf{z} = \begin{bmatrix} \delta^{10} \oplus \gamma\delta^{22} \oplus \gamma^{4}\delta^{30}(\gamma\delta^{6})^{*} \oplus \gamma^{10}\delta^{+\infty} \\ \varepsilon \end{bmatrix}$$

By convention, the first event is the number 0 and the trajectory of this should be interpreted as follows: 0 task should be done no later than 10 time, and the task 1, 2 and 3 at the latest during the 22. Then there is to be executed task 4, at 32 and then each one next task every 6 units of time. The final monomial $\gamma^{9}\delta^{+\infty}$ means that the task 8 is the last for this process. It also means that the task 9 and the next is not implemented (the term is infinite).

Calculation of optimal control is determined by (11)

$$\mathbf{u} = \begin{bmatrix} e \oplus \gamma\delta^{8} \oplus \gamma^{2}\delta^{10} \oplus \gamma^{3}\delta^{12} \oplus \gamma^{4}\delta^{20} \oplus \gamma^{5}\delta^{26} \oplus \gamma^{6}\delta^{32} \oplus \gamma^{7}\delta^{38} \oplus \gamma^{8}\delta^{44} \oplus \gamma^{9}\delta^{+\infty} \\ \varepsilon \\ \varepsilon \end{bmatrix}$$

$$\mathbf{y} = \begin{bmatrix} \delta^{10} \oplus \gamma\delta^{18} \oplus \gamma^2\delta^{20} \oplus \gamma^3\delta^{22} \oplus \gamma^4\delta^{30} \oplus \gamma^5\delta^{36} \oplus \gamma^6\delta^{42} \oplus \gamma^7\delta^{48} \oplus \gamma^8\delta^{54} \oplus \gamma^9\delta^{+\infty} \\ \varepsilon \end{bmatrix}$$

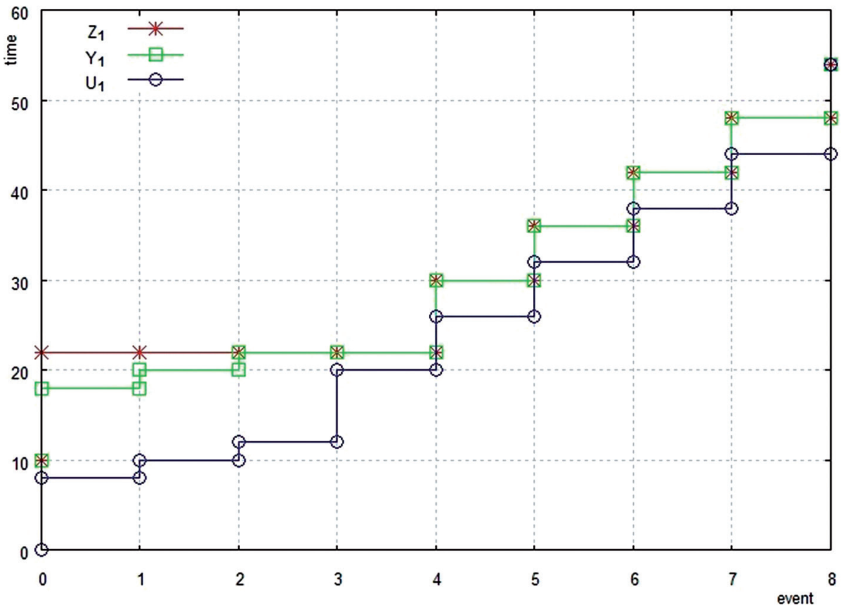


Fig. 8. Graphical representation of $z_1$, $y_1$ and $u_1$

Results as trajectories $z_1$, $y_1$ and $u_1$ are shown in Fig. 8. We can check that the optimal control **u** well meets the specification, i.e. that the output y is less than or equal to the **z**

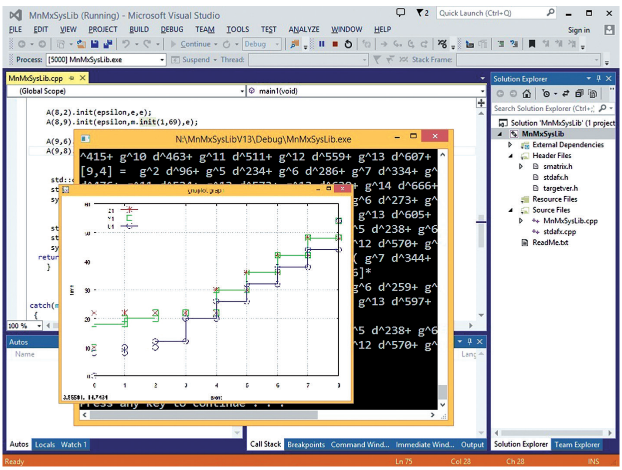


Fig. 9. VC++ platform of software tools for MPLS (in the implementation)

Calculations were performed and graphically presented using own software package currently developed on the Visual Studio 2013 platform (Fig. 9) with software library [28] and plot application Gnuplot [27].

## 7. Conclusions

In this article an overview of selected control structures and particular consideration of the control of the open MPLS has been presented. The main purpose is the synthesis of the control input, when we know global transfer **H** and reference (desired) input is updated. In the next step the problem of permissible deviations of real **H** should be elaborated, and the presence of uncontrollable input transitions. The problem formulated in the article has the close analogy with the problems encountered in classical control theory. There is not only feedback control but also predictive and robust control. There may be a need for effective control to use decoupling in multidimensional systems with cross-coupling interaction (like in computer control system [24]). Other problems concern different failures – events of data loss and damage while transmission. The solutions obtained do not completely eliminate the consequences of failures (i.e. delay), but are used for maintenance of the stability and elimination of memory overflow [23].

It is important to follow the new solutions and development of theoretical researches, concerning the classical theory of the system. It is planned to evolve practical applications and to create new or expand existing informatic tools. Further development of this software is planned.

## References


[1] Ästrom K., *Robustness of a design method based on assignment of poles and zeros*, IEEE Trans. on Automatic Control, 25, 588-591.

[2] Baccelli F., Cohen G., Olsder, G. Quadrat, J., *Synchronization and Linearity, An Algebra for Discrete Event Systems*, Wiley and Sons, 1992.

[3] Braker H., *Algorithms and Applications in Timed Discrete Event Systems*, PhD thesis, Delft University of Technology, 1993.

[4] Brunsch T., Hardouin L., Raisch J., *Modelling Manufacturing Systems in a Dioid Framework*, [in:] *Formal Methods in Manufacturing*, ed. Campos J., 29-94.

[5] Cohen G., Dubois D., Quadrat J., Viot M., *Analyse du comportement périodique des systèmes de production par la théorie des dioïdes*, Rapport de recherche 191, INRIA, Le Chesnay, France, 1983.

[6] Cohen G., Dubois, D. Quadrat J., Viot M., *A linear system theoretic view of discrete event processes and its use for performance evaluation in manufacturing*, IEEE Trans. on Automatic Control, AC–30, 1985, 210-220.

[7] Cohen G., Gaubert S., Quadrat J.P., *Linear Projectors in the max-plus Algebra*, Proceedings of the IEEE Med. Conf, Cyprus, Jul, 1997.

[8] Cohen G., Gaubert S. Quadrat J., *Max-plus algebra and system theory: Where we are and where to go now*, Annual Rev. Elsevier-IFAC, Vol. 23, No. 1, 1999, 207-219.

[9] Cohen G., Moller, P., Quadrat, J., Viot, M., *Algebraic Tools for the Performance Evaluation of Discrete Event Systems*, IEEE Proceedings, Special issue on Discrete Event Systems, 77(1), 1989, 39-58.
[10] Cottenceau B., *Contribution à la commande de systèmes à événements discrets, synthèse de correcteurs pour les graphes d'événements temporisés dans les dioïdes*, Thèse, LISA – Université d'Angers, 1999.
[11] Cottenceau B., Hardouin L., Boimond J.-L., Ferrier J.-L., *Model reference control for timed event graphs in dioids*, Automatica, vol. 37, 2001, 1451-1458.
[12] Cuninghame-Green R., *Minimax Algebra*, "Economics and Mathematical Systems", Springer, 1979.
[13] Davey B., Priestley H., *Introduction to Lattices and Order*, Cambridge University Press, 1990.
[14] Gruet B., *Structure de commande en boucle fermée des systèmes à événements discrets*, DEA, LISA – Université d'Angers – France, 1995.
[15] LeBoudec J.-Y., Thiran P., *Network Calculus*, Springer Verlag, 2002.
[16] Lhommeau M., *Etude de systèmes à événements discrets dans l'algèbre (max,+), 1. Synthèse de correcteurs robustes dans un dioïde d'intervalles. 2. Synthèse de correcteurs en présence de perturbations*, Thèse, LISA – Université d'Angers, 2003.
[17] Lhommeau M., Hardouin L., Cottenceau B., Jaulin L., *Interval analysis and dioid: application to robust controller design for timed event graphs*, "Automatica", vol 40(11), 2004, 1923-1930.
[18] Libeaut L., Loiseau J.-J., *On the control of timed event graphs*, [in:] Proceedings of WODES'96, Edinburgh, 1996.
[19] Lotito P., Mancinelli E., Quadrat J.-P., *A minplus derivation of the fundamental car-traffic law*, Report 324, 2001, INRIA.
[20] Maia C., Santos-Mendes R., Hardouin L., *Some Results on Identification of Timed Event Graphs in Dioid*, [in:] 11th IEEE Mediterranean Conference on Control and Automation, MED'2003, Rhodes, Grèce.
[21] Menguy E., Boimond J.-L., Hardouin L., Ferrier J.-L., *Just in time control of timed event graphs, update of reference input, presence of uncontrollable input*, "IEEE Transactions on Automatic Control", vol. 45, no. 11, 2000, 2155-2159.
[22] Raszka J., Jamroż L., *Max-Plus linear system in control of data processing*, "Technical Transactions. Fundamental Sciences", Y. 111, iss. 16, 2-NP, 2014.
[23] Raszka J., Jamroż L., *Reducing human resources in management of information technology (IT) projects*, HSI, IEEE, 2015.
[24] Raszka J., *Wspomagana komputerowo analiza stanów dynamicznych walcowni ciągłej blach walcowanych na zimno oparta na kompleksowym modelu matematycznym*, Biblioteka AGH, Kraków 1987.
[25] Schutter B.D., van den Boom T., *Max-plus algebra and max-plus linear discrete event systems: An introduction*, Proceedings of the 9th International Workshop on Discrete Event Systems (WODES'08), Göteborg, Sweden, pp. 36-42, May 2008.
[26] Hardouin L., *Sur la commande lineaire de systemes a evenements discrets dans l'algebre (max,+)*, Automatic. Universite d'Angers, 2004.
[27] Gnuplot homepage [on line], http://www.gnuplot.info (access: 01.07.2016).
[28] Software tools to handle periodic series in dioid http://perso-laris.univ-angers.fr/~hardouin/outils.html (access: 01.07.2015).