STUDIA UNIVERSITATIS

Revistă științifică a Universității de Stat din Moldova, 2011, nr.2(42)

2D VISUAL MEMBRANE PETRI NETS MODEL OF ALLELIC GENE REGULATION

Aurelia PROFIR

Catedra Tehnologii de Programare

În lucrare este propusă o nouă clasă de Rețele Petri, numită 2D Visual Membrane Petri Nets. În calitate de component de bază al rețelelor, a fost adăugată structura membranară 2D. În plus, este propusă o nouă extensie a formalismului expresiilor descriptive pentru o reprezentare adecvată a componentelor 2D Visual Membrane Petri Nets. În calitate de exemplu de P sisteme cu timp continuu, este analizat modelul P transducerului de reglare a expresiei genelor alele. Componentele modelului de P transducer sunt exprimate prin componentele 2D Visual Membrane Petri Nets și prin elemente descriptive.

I. Introduction

Membrane computing is an emergent branch of Natural Computing. P systems (called membrane systems) are abstract parallel and distributed computing devices inspired by the structure and the functioning of the living cell [1]. A lot of P systems formalisms had been proposed: cell-like and tissue-like P systems, discrete, continuous time and continuous P systems, etc. [1,2].

We introduced a continuous-time P systems formalism, a formalism of P transducer model with elementary membranes, arranged into chains, for elucidation both of discrete and continuous aspects of allelic gene structure-functional organization and the functioning of genetic machinery [3-5]. *DNA* (in genes) is mapped by means of systems of elementary membranes, arranged into chains, for capture gene structure and relevant functional aspects of temporal behavior of gene expression regulation.

Petri Nets (PN) are very popular formalism for the representation, modeling, validation, simulation and performance analysis of concurrent distributed systems. Various extensions have been made within the PN framework: discrete, timed stochastic, continuous, hybrid, etc. [6-8]. The interest of relating P systems with the PN models of computation lead to several important results on simulation and decidability issues. Some efforts have been made to simulate discrete P systems with Petri nets [9]. In [4,5] continuous-time P Systems are mapped through Descriptive Rewriting Timed Petri Nets. In section IV we introduce a new class of Petri Nets, named 2D Visual Membrane Petri Nets (*2D VMPN*), and it is shown that the components of continuous-time P Systems could be modelled by the components of *2D VMPN*. As an example of continuous-time P Systems, a P transducer model of allelic gene regulation is mapped by components of *2D VMPN*. A new extension of descriptive expression (*DE*) formalism is proposed to express the components of *2D VMPN* in section V. In the next sections a *2D VMPN* model of allelic gene regulation and the mapping of *2D VMPN* components by *DE* are analized.

II. Basic ingredients of continuos-time P systems

In this section we describe the basic ingredients of continuous-time P systems. P systems as discrete–time computing devices [1] have limited applicability to common biological data sets. Continuous–time P systems combine the discrete and continuous aspects of biological membrane systems and can easily describe both the discrete aspects of biosystems such as one-by-one molecular interactions and the continuous-valued rates of biochemical reactions. High-speed and slow biochemical reactions continuously and in parallel evolve in the living cell. Such timing hierarchy is an essential, distinctive feature of biosystems. Biochemical reaction rates can depend on the concentration of the reactants.

A notion of *rate*, denoted by the symbol $r_m \in \mathbf{R}_+$, $m \in \mathbf{N}_+$ associated with rules of P systems, is introduces in [3]. A rule is applied at an instant $t_{i+1} \in \mathbf{R}_+$ according to r_m as follows. We consider ***time delay***, denoted by d_m, which is inverse proportional to the rate r_m, between the instant $t_i \in \mathbf{R}_+$ when the rule became to be available and the instant t_{i+1} when the rule is applied. $d_m = t_{i+1} - t_i$.

A continuous-time P system is a construct:

$$\Pi = (V, \mu, w_1^0, \ldots, w_k^0, R_1, \ldots, R_k),$$

where:

- V is an alphabet of *objects*;
- μ is a membrane structure of the membrane system Π. $k \geq 1$ is the degree of the Π system (number of membranes);
- $w_1^0, \ldots, w_k^0$ are objects over V^* initially present in each membrane of the membrane structure μ;
- $R_1, \ldots, R_k$ are finite sets of rules associated with each membrane of the membrane system μ. The rates of biological processes, are denoted by $r_m \in \boldsymbol{R}_+$, $m \in \boldsymbol{N}_+$, and are indicated as superscripts associated with the rules of a Π system. All available rules are applied according to their rates (or delay times) in a non-deterministic maximally parallel manner.

Our aim is to express ingredients of continuous–time P systems [3,5] by means of *2D VMPN* components for the validation, simulation and performance analysis of P system models. To express the compartimentisation of membrane systems we introduce membrane structure, denoted by μ as a component of Discrete-Timed PN.

III. Discrete-Timed Petri Nets

In this section, we describe discrete-timed PN (TPN). *A discrete-timed* PN is a structure TPN= < *P*, *T*, *Pre*, *Post*, *Test*, *Inh, G, Pri*, K_p, M_0, θ >, where:

- *P* and *T* are finite sets of discrete *places* and discrete *transitions*. $P \cap T = \varnothing$.
- $\forall p \in P$, *Pre*, *Test*, $Inh: P \times T \to Bag(P)$ and $Post: T \times P \to Bag(P)$ are the forward, test, inhibition and backward functions in the multi-sets of *P*, which define the set of arcs A with marking-dependent multiplicity of the *input* and *output* arcs, connecting transitions and places. The set A i*s* partitioned into tree subsets: *direct normal arcs* A_d, *inhibitory* A_h and *test* A_t *arcs.* Default value of these arcs is 1.
- $G: T \times IN_+^{|P|} \to \{0,1\}$ is *guard function* for each transition as a Boolean function (default value is *true*);
- *Pri*: $T \to IN_+$ defines the priority functions for the firing of each transition;
- $K_p: P \to IN_+$ is the capacity of places, and by default being infinite value;

The complete marking (state) of a net is described by a vector–column $M = (m_i p_i, m_i \geq 0, \forall\, p_i \in P)$, where *M*: $P \to IN_+$ are marking functions of places. The initial marking of net is M_0. The set of transitions *T* can be partitioned into a set T_0 of immediate transitions and a set T_τ of timed transitions, $T = T_0 \cap T_\tau = \varnothing$. An immediate transition $t_j \in T_0$ is drawn as a black thin bar and a timed transition $t_k \in T_\tau$ – as a black rectangle, *Pri*(T_0) > *Pri*(T_τ);

- θ : $T \times \boldsymbol{N}_+ \to \boldsymbol{R}_+$ is a weight function that maps transitions into real numbers $\boldsymbol{R}_+$ (delay time or weight speeds). It can be marking dependent. The delay time $\theta(t_i,M)=d_i(M)$ defines the transitions firing duration for each timed transition of T_τ. If several timed transitions $t_j \in T(M)$ are enabled concurrently, than either fire in competition or independently. We assume that a *race condition* exists between them. Two or more enabled transitions will be executed in parallel mode. The degree of parallelism is determined by the number of enabled firing transitions with the same delay time.

IV. 2D Visual Membrane Petri Nets

Petri Nets are useful for modelling concurrent, distributed, asynchronous systems. New extensions for developing new formalisms to describe more appropriate, flexible and convenient dynamic systems with complex restructured structures are introduced [6, 7, 9, 11]. In order to represent the modular structure, the hierarchical organization of computational processes and their dynamics, using the concept of P systems [1], new extensions of Petri nets, called descriptive rewriting timed Petri nets, have been introduced [8]. To model membrane biological systems we propose a new extension of Petri nets, named 2D Visual Membrane Petri Nets.

In the section II the basic ingredients of P systems are described. The main ingredients of P systems are the membrane structures, delimiting compartments where multisets of objects evolve according to (reaction) rules of a bio-chemical inspiration. In Fig.1 a hierarchical organization of a membrane structure is depicted, where the membrane structure consists of 9 membranes; each membrane defines a compartment (region); the external compartment is called skin; compartment (or membranes) which do not contain other compartments are called elementary; A membrane can contain other membranes, and the rules can process both objects and membranes.

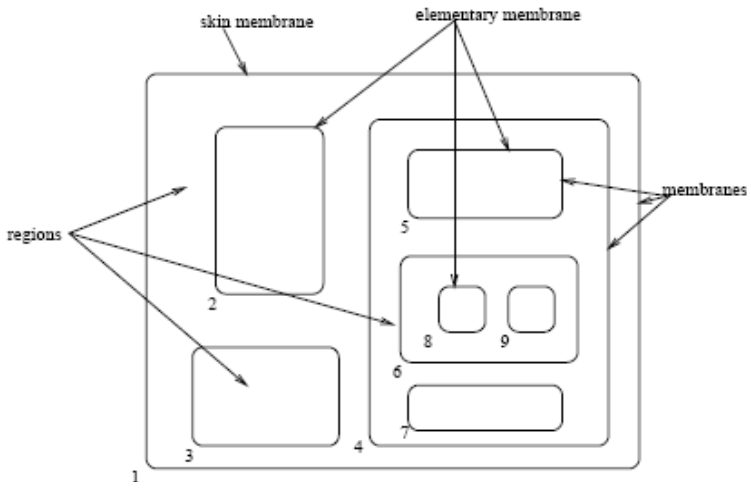

Figure 1. Grafical reprezentation of a hierarchical organization of a membrane structure. Membranes labelled with 2, 3, 5, 7, 8, 9 are elementary membranes.

Thus, membrane computing can be defined as a framework for devising cell-like or tissue-like computing models which process multisets in compartments defined by means of membranes [1]. Membrane computing models are (in general) distributed and parallel.

A 2D Visual Membrane PN system is a construct *2D VMPN* = < TPN, μ >, where: TPN is a discrete-timed PN considered above and μ is a membrane structure, which consists of 2D membranes.

All primitives of the *2D VMPN* formalism are shown in Figure 2.

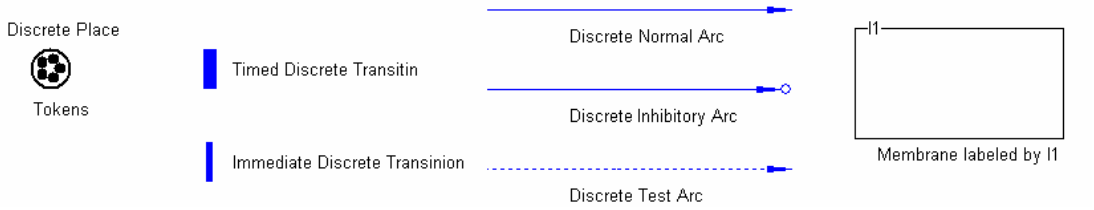

Figure 2. Primitives of the *2D MPN* formalism.

A membrane structure μ of VMPN model, consisting of 2D labeled membranes, maps a membrane structure μ of elementary membranes of a continuous-time P system model. Elementary membranes of continuous-time P systems models can be modeled by 2D labeled membranes of *2D VMPN*. The skin membrane of *2D VMPN* models (that maps the skin membrane of P systems models [1]) is usually not represented.

In Figure 3 a basic element of *2D VMPN* (*bMPN*) and its derivatives are represented.

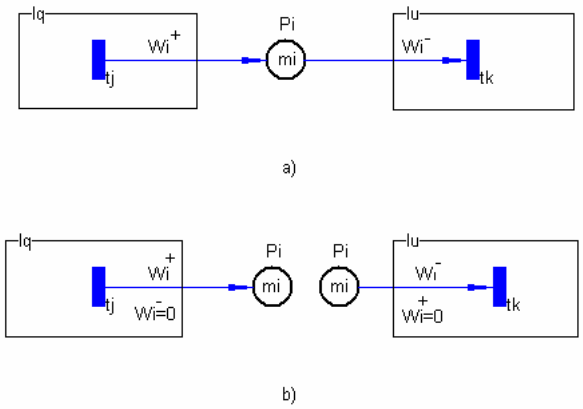

Figure 3. A basic element of *2D VMPN* (*bMPN*) (a) and its derivatives (b).

Within proposed formalism all components of the *2D VMPN* (transitions, locations and arcs) are localized in labeled membranes of the membrane structure μ of the model. In Figure 2 the input transion t_j for location p_i belongs to the membrane labeled by l_q; $t_k = p_i^{\bullet}$ is the output transition, located in membrane labeled l_u; the place p_i with an initial marking $m_i = M_0(p_i)$ is located in the membrane labeled by l_s, l_s is the label of the skin membrane of *2D VMPN* model (that corresponds to the *skin* membrane of the P transducer model) and this membrane is not represented here; the flow relation functions $W_i^+ = Pre(t_j, p_i)$ and $W_i^- = Post(p_i, t_k)$ return the multiplicities of the input and output arcs of the place p_i. We will consider that input and output arcs are located in membranes where places (with which they are associated) are located.

V. Descriptive Expressions of *2D MPN*

On the basis of the concept of a basic descriptive expression element (*bDE*), introduced in [13, 5] we propose a new extension of *DE* formalism for analytical representation of a basic element of *bMPN*. In this section we propose a following *DE*: $bDE = {}^{l_q}|_{t_j}^{d_j}\ m_i^{\,l_s} p_i \left[W_i^+, W_i^- \right]^{l_u}|_{t_k}^{d_k}$, where $t_j = {}^{\bullet}p_i$ is the input transition for p_i location, localized in the membrane labeled by l_q; $t_k = p_i^{\bullet}$ is the output transition, located in membrane labeled l_u, with d_j and d_k are delay times of the transitions t_j and t_k, located into the membrane labeled by l_q and l_u, respectively; the place p_i with an initial marking $m_i = M_0(p_i)$ is located in the membrane labeled by l_s (skin membrane of the model); the flow relation functions $W_i^+ = Pre(t_j, p_i)$ and $W_i^- = Post(p_i, t_k)$ return the multiplicities of the input and output arcs of the place p_i. We will consider that input and output arcs are located in membranes where places are located (Figure 3).

The derivative elements of *bDE* are: for $p_i^{\bullet} = \varnothing,\ W_i^- = 0$ is ${}^{l_s}|_{t_j}^{d_j}\ m_i^{\,l_s} p_i [W_i]$ with the output place p_j of t_j and for ${}^{\bullet}p_i = \varnothing,\ W_i^+ = 0$ is $m_i^{\,l_s} p_i W_i^{\,l_u}|_{t_k}^{d_k}$ with the input place p_i of t_k. If the initial marking m_i of the place p_t is zero, we can omit m_i=0 in *bDE*. If $W_i^+ = W_i^- = 1$, we present *bDE* and its derivatives as following (see Figure 3(b)):

$${}^{l_q}|_{t_j}^{d_j}\ m_i^{\,l_s} p_i^{\,l_u}|_{t_k}^{d_k},\ {}^{l_q}|_{t_j}^{d_j}\ m_i^{\,l_s} p_i \text{ or } m_i^{\,l_s} p_i^{\,l_u}|_{t_k}^{d_k}.$$

We can build more complex descriptive expressions (*DE*) using composition operations.

Definition 1: DE of composite PN is either a *bDE* or a composition of *DE*: $DE ::= bDE \mid DE * DE \mid \circ DE$, where $*$ represents any binary composition operation and $\circ$ is any unary operation. ■

The composition operations are reflected at the level of the DE components of *2D MPN* models by fusion of places, fusion of transitions with same names or sharing of subnets.

Place-Sequential Operation: This binary operation, denoted by " | ", named *sequential operator*, determines the logics of an interaction between two local states*: p* (pre-condition) and p_i (post-condition), by the action t_i. They are in precedence and succeeding (causality-consequence) relation. Sequential operator is one of the *basic mechanisms* to build the PN models. The sequential operation possesses an *associative*, *reflexive* and *transitive* properties, but is *not commutative*.

$DE1 = m_i^{\,l_s} p_i \left[W_i\right]^{l_u}|_{t_j}^{d_j}\ m_k^{\,l_s} p_k \left[W_k\right] \neq m_k^{\,l_s} p_k \left[W_k\right]^{l_u}|_{t_j}^{d_j}\ m_i^{\,l_s} p_i \left[W_i\right]$ means that the specified conditions (local state), associated with place-symbol p_i, are fulfilled always. *DE1* takes place before then the occurrence of the conditions, associated with place-symbol p_k, and the action t_j, are happened.

The PN modeling of the *iteration* operation is obtained by the fusion of the head (input) place with the tail (output) place that have the same names (*closing* operation). $DE2 = m_i^{\,l_s} \widetilde{p}_i W_i^{\,l_u}|_{t_k}^{d_k}$, "~" is the *test operator* represents the *test arc*.

- *Inhibition Operation:* This unary operation is represented by *inhibitory operator* " $^-$ " and $DE3 = m_i^{\,l_s} \overline{p}_i W_i^{\,l_u}|_{t_k}^{d_k}$ describes the *inhibitor arc* in *2D MPN* models with a weight function (arc multiplicity).
- *Synchronization Operation:* This binary operation represented by "$\bullet$" or "$\wedge$", named *join* operator,

describes the rendezvous synchronization (by the transition t_j) of two or more conditions, represented by symbol-place $p_i \in {}^\bullet t_j$, $i = \overline{1,n}$. It indicates that all preceding conditions of occurrence actions must be completed. This operation has commutative and associative properties.

- *Split* O*peration:* This binary operation represented by "◊", named *split* operator, determines the causal relations between the action t_j and its post-conditions: after completion of the preceding action t_j simultaneously several other post-conditions can occur in parallel. Split operation possesses commutative and associative properties.
- *Competing Parallelism Operation:* This compositional binary operation is represented by *"∨"*, competing parallelism *operator*, means that it can be applied over two nets: N_A with $DE_A = A$, and N_B with $DE_B = B$. The resulting net N_R with $DE_R = R$ can be represented by $DE_R = R = A \vee B$. Here the places with the same names fuse. The fused places will inherit the incidence arcs of the places *A* and *B*. This operation possesses *commutative* and *associative* properties.
- *Precedence Relations between the Operations*: a) the evaluation of operations in *DE* are applied left-to-right; b) an unary operation binds stronger than a binary one; c) the "•"operation is superior to "|" and "◊", in turn, they are superior to the "∨" operation. More details about enabling and firing rules, and functioning of PN can be found in [13].

Refinement in *2D VMTN* is the process by which the macronodes (macroplaces and/or macrotransitions) labeled by *x* are replaced by new subnets. The refinement, specified by the *2D VMTN* modeler allows the refining of marking-controlled macronodes for every enabled binding in suitable time. In order to preserve the uniqueness of names, in the refining net, the nodes of the subnet can be prefixed by the name of the refined macronode. For instance, if the place p_i or the transition t_j belongs to the subnet that refines the macronode *x*, its name can be denoted as $x.\ p_i$ or $x.\ t_j$, respectively.

Otherwise, it can be denoted by distinguish name relative to the whole PN model. We shall employ marking-controlled refinement as a mechanism for implementing the substitution process, specified by the rewriting rule r_x. This enables the refinement of macronode *x* by a well-formed subnet represented by the descriptive expression DE_x in the form $x \triangleright DE_x$.

VI. Encoding of the ingredients of continuous-time P Transducer throughout the components of *2D VMPN*

The components of continuous-time P systems can be expressed by components of *2D VMPN* as follows. To every set of objects *a* associated with a region of the membrane structure of the P transducer model we put into correspondence a place located into a membrane l_i of *2D VMPN*, (Tab. II), where l_i is the label of membrane where the place is localized. The marking of the place represents the number of copies of the object *a*. The initial state of the continuous-time P system is represented through the initial marking M_0 of *2D VMPN*. Compartimentisation of the membrane structure of P transducer is reflected by membranes, where places and transitions of *2D VMPN* are located.

We remark two distinct situations connected to the mapping of rules of the P transducer into transitions of *2D VMPN* models:

- for every rule of the P transducer models can be put into correspondence one distinct transition t = (l_j, r)∈ T, where l_j is the label of membrane with which the rule r is associated [3,5];
- some rules of the P transducer can be modelled by two or more transitions (macronodes) (Table III).

VII. Allelic gene model of dominance

Let us analyse the genetic regulatory system of allelic genes linked as a regulatory network (with positive control of gene expression).

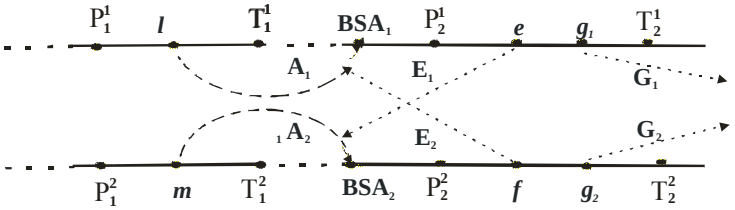

Figure 4. The scheme of molecular regulatory mechanism of alternative expression of allelic genes.

Figure 4 depicts the elements of the regulatory system of one pair of allelic genes, denoted by g_1 and g_2 that code for output signals G_1 and G_2 responsible for the observed inherited trait. Regulatory regions of the *e* and *f* genes are represented as BSA_1, P_2^1 and BSA_2, P_2^2. The *l* and *m* regulatory genes code for the A_1 and A_2 activator molecules, respectively. The *e* and *f* genes are under the positive control of the activator molecules and code the E_1 and E_2 regulatory enzymes, respectively. Due to their proteolytical activity, they destroy activator molecules of the opposite homologous chromosome. So, allelic genes are expressed in two alternative regimes.

We take into consideration that the rates of the enzymatic reaction can depend on different factors (for instance, temperature), considered as input signals. The enzymes can be either activated (or inactivated), i.e., the rates of proteolytical reaction of E_1 and E_2 enzymes can increase (or decrease) as a result of input signals action. So, they can (or not) destroy the A_1 or A_2 activators of the opposite chromosome. Using the P transducer concept we built a formal model which allows to illustrate the direct influence of temperature on allelic gene expression.

VIII. P transducer model of dominance

The genetic mechanism of the allelic gene switching controlled by different factors (i.e., input ``marked`` objects read from "input tape"), such as: the temperature, concentration of specific molecules, is modelled [3, 5]. The G_1 and G_2 proteins, designed as output ``marked`` objects, are sent out to the "output tape". Other factors which influence to functioning of genetic system (i.e., rates of biochemical reactions) are considered as environmental stimuli, denoted by b symbol. The writing of ``marked`` objects on the ``output tape`` is associated with the cell response to input factors.

A P transducer model of dominance illustrating the molecular mechanism of allelic genes expression, controlled by regulatory enzymes, may be represented in the following way:

$\Pi = (O_l, O_m, O_e, O_f, V, \mu_l, \mu_m, \mu_e, \mu_f, \mu_{cell}, w_{l(i)}, w_{m(i)}, w_{e(i)}, w_{f(i)}, w_1, E, R_{l(i)}, R_{m(i)}, R_{e(i)}, R_{f(i)}, R_1, R_2)$, $i=1,2,3$, where:

- O_l, O_m, O_e, O_f are alphabets of objects associated with systems of elementary membrane that map the l, m, e and f genes, respectively. $O_l = \{p, p_l\}$, $O_m = \{p, p_m\}$, $O_e = \{A_1, p, p_e\}$, $O_f = \{A_2, p, p_f\}$, where: A1 and A2 represent activator molecules, encoded by the l and m genes, respectively; p – RNA polymerase molecule; p_l, p_m, p_e, p_f, – RNA polymerase transcribing the l, m, e or f gene into mRNA copies;
- V is the alphabet of objects of the cell-like membrane system.
  $V = \{p, p_l, p_m, p_e, p_f, u_l, u_m, u_e, u_f, u, m, v, f, A_1, A_2, E_1, E_2, G_1, G_2, t, b\}$; $O_l, O_m, O_e, O_f \subseteq V$,
  where:
- u_l, u_m, u_e, u_f represent nuclear copies of the l, m, e and f genes, respectively; E_1, E_2 – regulatory enzymes, encoded by the e1 and e2 genes; G1, G2 – output signals, encoded by the g1 and g2 genes; u, m, v, f – mRNA copies of the l and m genes; v and f – mRNA copies of the e1& g1 and e2& g2 genes, respectively; t – temperature (input signals); b – set of objects in the environment;
- $\mu_l, \mu_m, \mu_e, \mu_f$ are elementary membranes, arranged into continuous chains that map the l, m, e and f genes.
- $\mu_l = [_{l(1)} \quad _{l(1)}|_{l(2)} \quad _{l(2)}|_{l(3)} \quad _{l(3)}]$ is the elementary membrane structure, which represent the l gene. The direct communication between membranes along the chain is done in a one-way manner: $l(1) \rightarrow l(2) \rightarrow l(3)$. The first membrane of the chain (labelled l(1)) represents the regulatory region of this gene, it is called the input/output membrane. The last membrane of the chain representing the gene transcriptional termination site is named the output membrane. The gene-coding region is mapped by the elementary membrane labelled l(2). The membrane structures μ_m, μ_e, μ_f of the m, e and f genes are similar to the membrane structure μ_l of the l gene;
- $\mu_{cell} = [_2 \, [_1 \quad]_1 \,]_2$ represents the cellular membranes. Cellular membrane (the skin) is represented by membrane 2, nucleus envelope – by membrane 1;
- $w_{l(i)}$, $w_{m(i)}$, $w_{e(i)}$, $w_{f(i)}$, $(i = 1, 2, 3)$ represent the initial sets of objects over O_l, O_m, O_e, O_f, associated with each elementary membrane of the membrane structures $\mu_l, \mu_m, \mu_e, \mu_f$, respectively.

- w_1 is the initial multiset over V* associated with the region delimited by the membrane 1 (which represents the cell nucleus).
- The initial configuration of the P transducer can be represented as following:
- $w_{l(2)} = \{p_l\};\ w_{m(1)} = \{p\};\ w_{m(3)} = \{p_m\};\ w_{e(1)} = \{A_1, p\};\ w_{e(2)} = \{p_e\};\ w_{f(2)} = \{p_f\};$
  $w_1 = \{p, p_l, p_m, p_e, u_l, u_m, u_e, u, m, v, A_1, E_1, G_1, b\};$
- $E \in V$ is the set of objects in the environment. $E = \{t, b, G_1, G_2\}$;
- $R_{l(i)},\ R_{m(i)},\ R_{e(i)},\ R_{f(i)}, (i = 1, 2, 3)$ are finite sets of rules associated with each elementary membrane of the membrane structures $\mu_l, \mu_m, \mu_e, \mu_f$, respectively (Table 1). All the relevant molecular interactions one–by–one with DNA are modelled by these rules. The rates of rule application are indicated as superscripts associated with the rules.

All rules are applied in a non-deterministic maximally parallel manner. Only one object is specified in all rules described below (Table 1).

Table 1

Rules of the P transducer model of dominance

P systems		P systems	
M	*Rules*	*M*	*Rules*
l(1)	$(p, in)\|^{r_1}$	m(1)	$(p, in)\|^{r_5}$
	$(p \to (p_l, go))\|^{r_2}$		$(p \to (p_m, go))\|^{r_6}$
l(2)	$(p_l, go)\|^{r_3}$	m(2)	$(p_m, go)\|^{r_7}$
l(3)	$(p_l, out)\|^{r_4}$	m(3)	$(p_m, out)\|^{r_8}$
e(1)	$(A_1, in)\|^{r_9}_{A_1}$	f(1)	$(A_2, in)\|^{r_{15}}_{A_2}$
	$(A_1, out)\|^{r_{10}}$		$(A_2, out)\|^{r_{16}}$
	$(p, in)\|^{r_{11}}_{pA_1}$		$(p, in)\|^{r_{17}}$
e(2)	$(p \to (p_e, go))\|^{r_{12}}$	f(2)	$(p \to (p_f, go))\|^{r_{18}}$
e(3)	$(p_e, go)\|^{r_{13}}$	f(3)	$(p_f, go)\|^{r_{19}}$
	$(p_e, out)\|^{r_{14}}$		$(p_f, out)\|^{r_{20}}$
1	$(p_l \to (pu_l))\|^{r21}$	1	$(p_m \to (pu_m))\|^{r_{23}}$
	$(u_l, out)\|^{r_{22}}$		$(u_m, out)\|^{r_{24}}$
	$(p_e \to (pu_e))\|^{r25}$		$(p_f \to (pu_f))\|^{r_{27}}$
	$(u_l, out)\|^{r_{26}}$		$(u_f, out)\|^{r_{28}}$
	$(A_1^5 \to A_1^4)\|^{r29}$		$(A_2^5 \to A_2^4)\|^{r_{30}}$
	$(E_1^5 \to E_1^4)\|^{r31}$		$(E_2^5 \to E_2^4)\|^{r_{32}}$
	$(A_2E_1 \to E_1)\|^{r33}$		$(A_1E_2 \to E_2)\|^{r_{34}}$
2	$(b, read), in))\|^{r_{35}}$	2	$(m \to mA_2)\|^{r_{39}}$
	$(u \to uA_1)\|^{r_{36}}$		$(A_2, out_1)\|^{r_{40}}$
	$(A_1, out_1)\|^{r_{37}}$		$(m \to \lambda)\|^{r_{41}}$
	$(u \to \lambda)\|^{r_{38}}$		$(f \to fE_2G_2)\|^{r_{45}}$
	$(v \to vE_1G_1)\|^{r_{42}}$		$(E_2, out_1)\|^{r_{46}}$
	$(E_1, out_1)\|^{r_{43}}$		$(f \to \lambda)\|^{r_{47}}$
	$(v \to \lambda)\|^{r_{44}}$		$((G_2, write), out)\|^{r_{49}}$
	$(u \to \lambda)\|^{r_{38}}$		
	$((G_1, write), out)\|^{r_{48}}$		

Each rule of the P transducer has one of the following forms:

- $(x,in)|_{a}^{r_m}$ (or $(x,in)|_{ab}^{r_m}$) means that from the environment the object *x* enters the input/output membrane only in the absence of the object *a* (in the absence of the object *a* and in the presence of *b*). These conditional rules reflect the functional organization of the gene regulatory region;
- $(x,out)\,|^{r_m}$ means that the object *x* is sent *out* from the output membrane with which the rule is associated;
- $(x,go)|^{r_m}$ means that the object x leaves the membrane with which the rule is associated and passes to the next membrane of the chain;
- $(x \to (y,go))\,|^{r_m}$ means that the object *x* leaves the input/putput membrane, it is replaced by the object *y* that passes to the next membrane of the chain.

These rules model three stages of the gene transcription process. Rules, associated with the input/output membrane, model the initiation of transcription: RNA polymerase and transcription factors (reversibly) interaction with the regulatory region of gene controlling the gene expression process. Rules, associated with the membranes that map the gene-coding region, model the movement of RNA polymerase along the DNA. Rules, associated with the output membrane, model the RNA polymerase leaving the end of gene;

- R_1, R_2 are finite sets of rules associated with the regions delimitated by the skin membrane of the cell-like membrane system (that map the cytoplasm) and the membrane *1* associated with the nuclear membrane (Table 1). Biochemical reactions that evolve in the cell cytoplasm (a part of cellular protein network) are modelled by evolution rules. The objects *u* and *v* evolve according to given *evolution rules* of the form $u \to v$, meaning that *u* is replaced by *v* [1]. The action of input signals on the living cell and the cellular response are modelled by the *symport* rules associated with the skin membrane of the P transducer [5] (a part of cellular membrane network). All rules are applied in a non-deterministic maximally parallel manner.

IX. Mapping of rules of the P transducer model of dominance by DE

First of all we put into correspondence to every set of objects localized within a region delimited by an elementary membrane of P transducer model one place, localized into a membrane of *2D VMPN* model (Table 2).

Table 2

Corresponce of P transducer objects to *2D VMPN* places

P transducer		***2D VMPN***	
Elementary membranes	*Objects*	*Membrane labels*	*Places*
l(1)	p	l_1	p1
l(2)	p_l	l_2	p2
l(3)	p_l	l_3	p3
e(1)	A_1	e_1	p7
	p		p8
e(2)	p_e	e_2	p9
e(3)	p_e	e_3	p10
1	p	1	p15
	p_l		p16
	u_l		p17
	p_e		p20
	u_e		p21
	A_1		p24
	A_2		p25
	E_1		p26
2	b	2	p29
	u		p30
	A_1		p31
	v		p34
	G_1		p35
	E_1		p36

The mapping of a part of rules of the P transducer model of dominance by DE is presented into the Table 3.

Table 3

Mapping of a part of P transducer rules by DE

P transducer		***DE***	
M	*Rules*	*M*	*DEs*
e(1)	$(A_1, in)\|_{A_1}^{r_3}$	e_1	$1000^1 p_{24}{}^{e_1}\|_{t_9}^{d_9} 1^{e_1} p_7, g_9 = (m_7 = 0)$
	$(A_1, out)\|^{r_4}$		$1^{e_1} p_7{}^{e_1}\|_{t_{10}}^{d_{10}} 1000^1 p_{24}$
	$(p, in)\|_{pA_1}^{r_{11}}$		$157^1 p_{15}{}^{e_1}\|_{t_{11}}^{d_{11}} 1^{e_1} p_8,\ g_{11} = (m_8 = 0) \& (m_7 = 0)$
	$(p \to (p_e, go))\|^{r_{12}}$		$1^{e_1} p_8{}^{e_1}\|_{t_{12}}^{d_{12}} 3^{e_2} p_9,\ 3^{e_2} p_9 \triangleright \bigcup_{i=46}^{49} {}^{e_1} p_i {}^{e_1}\|_{t_{i+6}}^{d_i} {}^{e_1} p_{i+1}$
e(2)	$(p_e, go)\|^{r_{13}}$	e_2	$3^{e_2} p_9{}^{e_2}\|_{t_{13}}^{d_{13}} 1^{e_3} p_{10}, d_{13} = \sum_{i=52}^{55} d_i, 3^{e_2} p_9 \triangleright \bigcup_{i=46}^{49} {}^{e_2} p_i \|_{t_{i+6}}^{d_i} {}^{e_2} p_{i+1}$
e(3)	$(p_e, out)\|^{r_{14}}$	e_3	$1^{e_3} p_{10}{}^{e_3}\|_{t_{14}}^{d_{14}} {}^1 p_{20}$
1	$(p_e \to (pu_e))\Big\|^{r_{25}}$	1	${}^1 p_{20}{}^1\|_{t_{25}}^{d_{25}} {}^1 p_{21}$
	$(u_e, out)\Big\|^{r_{26}}$		${}^1 p_{21}{}^1\|_{t_{26}}^{d_{26}} {}^2 p_{34}, {}^2 p_{34} \triangleright \bigcup_{i=62}^{64} {}^1 p_i {}^1\|_{t_{i+2}}^{d_i} {}^1 p_{i+1}$
	$\left(A_1^5 \to A_1^4\right)\Big\|^{r_{29}}$		$1000^1 p_{24}{}^1\|_{t_{29}}^{d_{29}}$
	$\left(E_1^5 \to E_1^4\right)\Big\|^{r_{31}}$		${}^1 p_{26}{}^1\|_{t_{31}}^{d_{31}}$
	$\left(A_2 E_1 \to E_1\right)\Big\|^{r_{33}}$		$1000^1 p_{25} \bullet 1^1 \widetilde{p}_{26}{}^1\|_{t_{33}}^{d_{33}}$
2	$(b, read), in))\|^{r_{35}}$	2	$1 p_{28}{}^2\|_{t_{35}}^{d_{35}} {}^2 p_{29}$
	$(v \to vE_1G_1)\|^{r_{42}}$		${}^2 \widetilde{p}_{34}{}^2\|_{t_{42}}^{d_{42}} {}^2 p_{36} \Diamond {}^2 p_{35},\ {}^2 p_{34} \triangleright \bigcup_{i=62}^{64} {}^2 p_i {}^2\|_{t_{i+2}}^{d_i} {}^2 p_{i+1}$
	$(E_1, out_1)\|^{r_{43}}$		${}^2 p_{36}{}^2\|_{t_{43}}^{d_{43}} 1^1 p_{26}$
	$((G_1, write), out)\|^{r_{48}}$		${}^2 p_{35}{}^2\|_{t_{48}}^{d_{48}} p_{37}$,

For analytical representation we use the concept of descriptive expression formalism [13, 5] and for each element of *2D VMPN* we put into correspondence a descriptive expression *DE* (see Table 3).

In Table II to every set of objects localized within a region delimited by an elementary membrane of P transducer model corresponds one place, localized into a membrane of *2D VMPN* model.

X. Simulation results

Using the P transducer concept we built a *2D VMPN* model that allows us to illustrate the influence of temperature on allelic gene expression (Figure 5).

In the model of P transducer of the living cell [3,5] the number of elementary membranes that map the gene-coding region correlates with the maximal number of RNA polymerases transcribing the activated gene. Using of macronodes for mapping of some rules of P systems (see Table 3) we optimized the P transducer model of the living cell, representing the gene–coding region only by one elementary membrane. On the other hand, using the macronodes we can obtain more accurate simulation results.

In the model of P transducer of the living cell [3,5] the number of elementary membranes that map the gene-coding region correlates with the maximal number of RNA polymerases transcribing the activated gene. Using of macronodes for mapping of some rules of P systems (see Table 3) we optimized the P transducer

model of the living cell, representing the gene–coding region only by one elementary membrane. On the other hand, using the macronodes we can obtain more accurate simulation results.

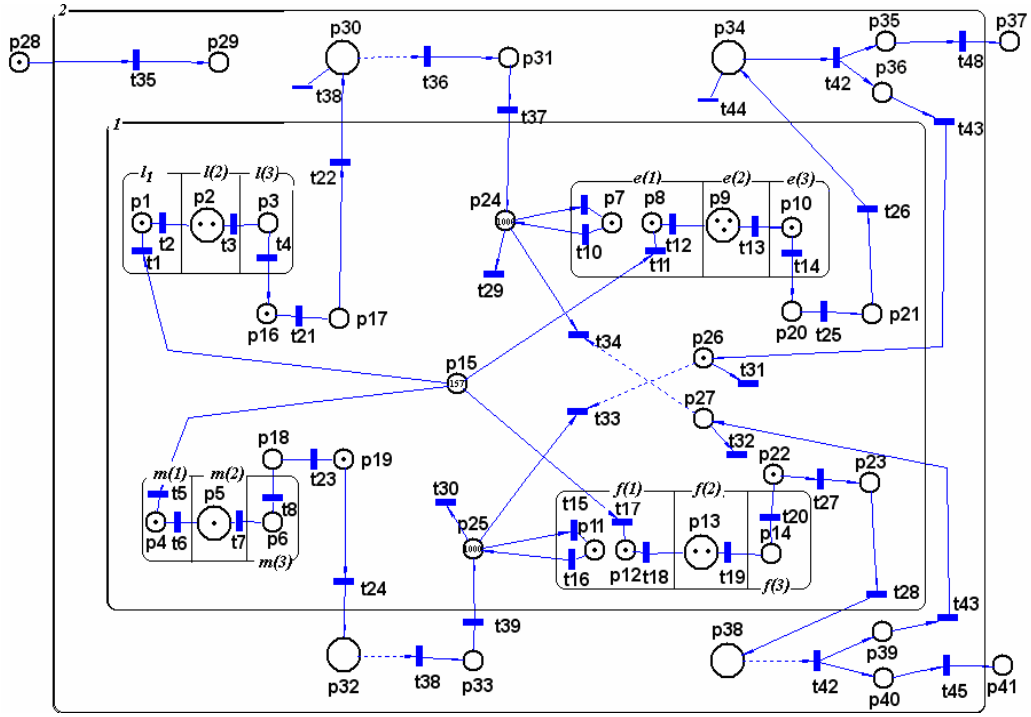

Figure 5. Screen snapshot of the *2D VMPN* model of the P transducer model of dominance.

In the model of P transducer of the living cell [3,5] the number of elementary membranes that map the gene-coding region correlates with the maximal number of RNA polymerases transcribing the activated gene. Using of macronodes for mapping of some rules of P systems (see Table 3) we optimized the P transducer model of the living cell, representing the gene–coding region only by one elementary membrane. On the other hand, using the macronodes we can obtain more accurate simulation results.

In 2D VMPN model of the P transducer the places *p2, p5, p9, p13, p30, p32, p34, p38* are mactonodes (macroplaces). For instance, to model the rule $(p_e, go)|^{d_{13}}$, associated with the region delimited by the elementary membrane *e(2)*, using the DE (Table 3) we used the macroplace *p9* ($3^{e_2} p_9 \triangleright \bigcup_{i=46}^{49} {}^{e_1} p_i {}^{e_1}|_{t_{i+6}}^{d_i} {}^{e_1} p_{i+1}$).

The number of places that composes the subnet, which replaces the macroplace *p9*, correlates with the number of *RNA* polymerases transcribing the activated gene. $d_{13} = \sum_{i=52}^{55} d_i, m_9 = \sum_{i=46}^{50} m_i.$

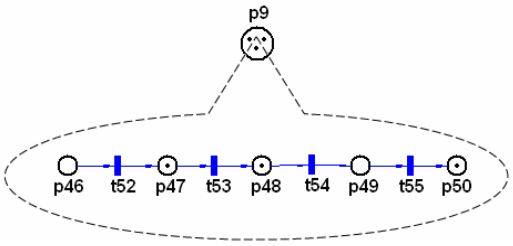

Figure 6. The subnet that replaces the macronode *p9.*

Important characteristics of gene function such as differential gene expression, functional aspects of dynamic behaviour of gene expression regulation are illustrated through *2D VMPN* simulation. Via the *2D VMPN* computational model we obtained the graphical representation of influence of temperature on allelic gene expression (Figure 7).

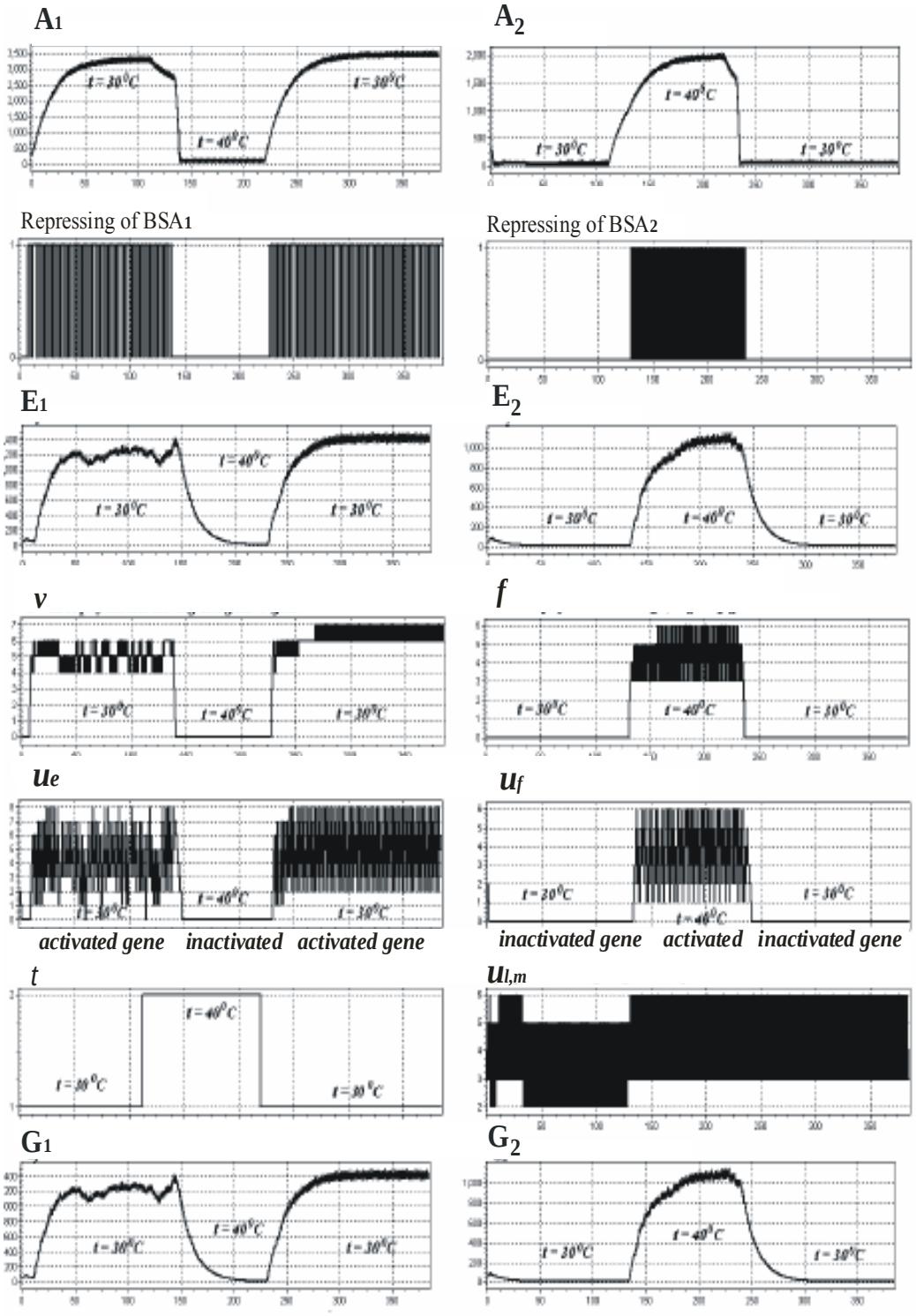

Figure 7. *2D VMPN* simulation of the temperature dependent switching of the allelic genes.

We investigated the case when some rules both of discrete and continuous-time P systems can be modelled by descriptive macronodes – the concept introduced in this paper. A simulator for *2D VMPN* models is implemented [12]. Using macronodes for the mapping of some rules of P systems we optimized the P transducer model of the living cell, representing the DNA (in a gene) only by three elementary membranes arranged onto chain. As consequence, we can obtain more accurate simulation results.

Due to the fact that cells self-regulate their biochemical activities, they can adapt to different conditions,

responding to stimuli. In this paper the allelic gene network is modeled by means of membrane systems to elucidate all the relevant molecular interaction one-by-one with *DNA*, all functional aspects of temporal behavior of allelic gene expression regulation.

Figure 7 illustrates that the temperature increasing transforms the gene f (e) from a dominant (a recessive) gene to a recessive (a dominant) one. The simulation is performed using *2D VMPN* parallel software tool for visual representation, formalization and simulation of *2D VMPN*-models. This tool allows verification and validation of behavioural properties of membrane systems and their visual interactive discrete-continuous simulation [14].

2D VMPN is a window-based, object-oriented parallel software tool, in which elements typical of hybrid Petri net models (discrete-continuous places, transitions, arcs, etc.) are manipulated under the assistance of basic syntactical rules that prevent the construction of incorrect models.

Our allelic gene regulation model is based on the principles of Systems biology that provides system integration of all components of hierarchical structure-functional organization of the living cell as an entire system.

Conclusions

A new class of Petri Nets, namely 2D Visual Membrane Petri Nets, is described in this paper. We shown that the components of continuous-time P systems could be modelled by the components of *2D VMPN* (with membrane structure μ, which was added as a basic component of the structure of the new proposed class of Petri Nets). As an example of continuous-time P systems, a P transducer model of allelic gene regulation is mapped by components of *2D VMPN*. A new extension of descriptive expression (DE) formalism is proposed to express the components of *2D VMPN*.

We analyzed the opportunity of simulation of continuous-time P transducer model using the *2D VMPN* components to obtain a model of computation leading to important results on simulation and decidability issues. Continuous-time P systems can be reduced to discrete ones when all rates (delay times) associated with rules of the P systems model will be equal to 1 (in discrete P systems representing one computation step). It is shown that continuous-time P system models can be simulated by *2D VMPN*, while the discrete P systems (a particular case of the continuous-time P systems) can be simulated using discrete PN [12].

Petri nets with membrane structures open up new perspectives for modelling biological membrane systems and offer new features for parallelization of membrane RP applications and running them on clusters of computers [14].

References:

1. Paun Gh. Membrane Computing. An Introduction. – Natural computing Series. Ed. G. Rozenberg, Th. Back, A.E. Eiben, J.N. Kok, H.P. Spaink. Leiden Center for Natural Computing, Springer – Verlag Berlin Heidelberg New York, 2002, p.420.
2. Ciobanu G., Paun Gh. and Stefanescu Gh.: P transducers // New Generation Computing 24(1) (2006), 1-28.
3. Barbacari N., Profir A., Zelinschi C. Gene regulatory network modelling by means of membrane Systems. – In: Proc. WMC6 2005, Wien, Austria, July 18-21, 2005, p.162-178.
4. Profir A., Gutuleac E., Boian E. Simulation of Continuous-time P Systems using Descriptive Rewriting Timed Petri Nets, Proc. SYNASC, IEEE Computer Science, 2005, p.458-461.
5. Profir A., Gutuleac E., Boian E. P transducer model of allelic gene network regulation. Proc. 8th International Conference on Development and Application Systems, DAS 2006, 25th - 27th May 2006, Suceava, Romania, p.405-412.
6. Llorens M., Oliver J. Marked-Controlled Reconfigurable Workflow Nets. In SYNASC, IEEE computer Society, 2006, p.407-413.
7. Guțuleac E. Descriptive Timed Membrane Petri Nets for Modelling of Parallel Computing / International Journal of Computers, Communications & Control, No: 3, Vol. I, Agora University Editing House. - Oradea, România, 2006, p.33-39.
8. Guțuleac E. Encoding P-systems with descriptive membrane timed Petri nets. Proceedings of the 4-rd International Conferance ICMCS-05, Vol. II, September 15-17, 2005, Chişinău, 2005, p.248-252.
9. Ajmone-Marsan M., Balbo G., Conte G., Donatelli S., and Francheschinis G. Modeling with Generalized Stochastic Petri Nets, ser. In Parallel Computing. - New York: Wiley, 1995.

10. Alla A., David H. Continuous and hybrid Petri nets // Journal of systems Circuits and Computers, 1998, 8(1), p.159-188.
11. Dal Zilio S., Formenti E. On the Dynamics of PB System: a Petri Net View. – In: Proc. WMC 2003, Lecture Notes in Computer Science 2933, Springer-Verlag, 2004, p.153-167.
12. Qi, Z., You J. and Mao H. P Systems and Petri Nets. – In: Proc. WMC 2003, Lecture Notes in Computer Science, vol. 2933, Springer, Berlin, 2003, p.387-403.
13. Gutuleac E. Descriptive Compositional Construction of GSPN Models for Performance Evaluation of Computer Systems. - In: Proceeding of 8-th International Symposium on Automatic Control and Computer Science, SACCS2004, Iasi, Romania, 2004, ISBN 973-621-086-3.
14. Profir A., Gutuleac E., Ciurganov D., Danu S. Modelling of molecular genetic mechanisms involved in the development of cardiovascular diseases using Visual Membrane Petri Nets, 1st National Symposium on e-Health and Bioengineering - EHB 2007, 16-17th November, 2007, Iasi, Romania, p.157-160.

Prezentat la 23.02.2011