Generalized Nets Modeling: A General Concept

Krassimir Atanassov[1] and Maciej Krawczak[2]

[1]Centre for Biomedical Engineering, Bulgarian Academy of Sciences
Acad. G. Bonchev Str., Bl. 105, Sofia-1113, Bulgaria
krat@argo.bas.bg
[2]Systems Research Institute, Polish Academy of Sciences
Newelska 6, 01-447 Warsaw, Poland
krawczak@ibspan.waw.pl

ABSTRACT

The paper consider the so-called generalized nets as a extansion of Petri nets. First the basic of the theory of generalized nets is introduced, next the algorithm of generalized nets is described. Algebraic aspects of generalized nets as well as operator aspects of generalized nets are described. At the end we showed, in a brief way, one possible application of generalized nets, namely for aggregation of neural networks.

Keywords: modeling, generalized nets, knowledge representation, system science.

1. INTRODUCTION

In 1982 K. T. Atanassov [1] proposed a new definition of nets for modelling and analysing various kinds of dynamic systems, the nets are called *generalized nets*.

In several papers it was shown that existing *Petri nets* were particular cases of generalized nets. The conception of generalized nets is based on developing a relation *place – transition*.

Generalized nets are characterized by:
- a *static structure*,
- dynamical elements called *tokens*,
- *temporal components*.

The static structure of generalized nets is characterized by *transitions*.

Tokens are described by changeable *characteristics*, and characteristics of tokens play a roll of *memory* of the nets.

There three global temporal constants: the initial moment in which the net starts functioning, the elementary time-step of the process, and the duration of functioning.

Generalized nets can be used for:
- comparing different types of nets as mathematical objects,
- investigating properties of generalized nets and transfer them to other nets,
- modeling in details real processes.

The theory of generalized nets, by analogy with the theory of Petri nets, can be divided into two basic fields - a *special* and a *general* theory of generalized nets.

The special theory of generalized nets concerns both the definitions and the properties of generalized nets as well as modifications of generalized nets.

The general theory of generalized nets deals with different aspects like: algebraic, logical, operation, program, methodological and topological.

In a book [1] one can find the basic elements of the theory of generalized nets, where generalized nets are defined as extensions of the ordinary Petri nets and their modifications.

A list of 353 scientific works related to generalized nets as a review and bibliography on generalized nets theory and applications can find in [3].

2. THE CONCEPT OF GENERALIZED NETS

The first basic difference between generalized nets and the ordinary Petri nets is the place – transition relation [1]. In the theory of generalized nets the transitions are objects of a very complex nature. The places are marked by ○, and the transitions by |. Generalized nets contain tokens, which are transferred from place to place. Every token bears some information, which is described by token's characteristic, and any token enters the net with an initial characteristic. After passing a transition the tokens' characteristics are modified.

The transition has input and output places, as it is shown in Figure 1

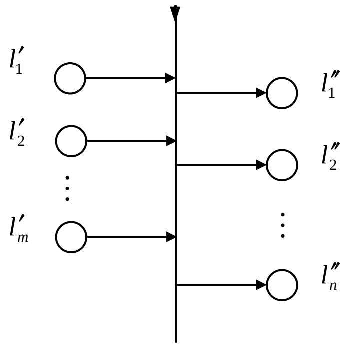

Figure 1. A generalized net transition.

Formally, every transition is described by a seven-tuple

$$Z = \langle L', L'', t_1, t_2, r, M, \square \rangle \qquad (1)$$

where:

- $L' = \{l'_1, l'_2, \ldots, l'_m\}$ is a finite non empty set of the transition's input places,
- $L'' = \{l''_1, l''_2, \ldots, l''_m\}$ is a finite non empty set of the transition's output places,
- t_1 is the current time of the transition's firing,
- t_2 is the current duration of the transition active state,
- r is the transition's *condition* determining which tokens will pass (or *transfer*) from the transition's inputs to its outputs; it has the form of an *index matrix* described in [4]

$r =$

	l''_1	$\cdots$	l''_j	$\cdots$	l''_n
l'_1	r_{11}	$\cdots$	r_{1j}	$\cdots$	r_{1n}
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$
l'_i	r_{i1}	$\cdots$	r_{ij}	$\cdots$	r_{in}
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$
l'_m	r_{m1}	$\cdots$	r_{mj}	$\cdots$	r_{mn}

  where r_{ij} is a predicate that corresponds to the i-th input and the j-th output places, $1 \le i \le m$, $1 \le j \le n$; when its truth value is *true*, a token is allowed to pass the transition from the i-th input place to the j-th output place,

- M is an index matrix of the capacities of transition's arcs:

$M =$

	l''_1	$\cdots$	l''_j	$\cdots$	l''_n
l'_1	m_{11}	$\cdots$	m_{1j}	$\cdots$	m_{1n}
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$
l'_i	m_{i1}	$\cdots$	m_{ij}	$\cdots$	m_{in}
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$
l'_m	m_{m1}	$\cdots$	m_{mj}	$\cdots$	m_{mn}

  where $m_{ij} \ge 0$ are natural numbers;

- $\square$ is an object of a form similar to a Boolean expression, it may contain as variables the symbols that serve as labels for transition's input places, and $\square$ is an expression built up from variables and the Boolean connectives $\wedge$ and $\vee$ whose semantics is defined as follows
  $\wedge(l_{i1}, l_{i2}, \ldots, l_{iu})$ - every place $l_{i1}, l_{i2}, \ldots, l_{iu}$ must contain at least one token, $\vee(l_{i1}, l_{i2}, \ldots, l_{iu})$ - there must be at least one token in all places $l_{i1}, l_{i2}, \ldots, l_{iu}$, where $\{l_{i1}, l_{i2}, \ldots, l_{iu}\} \subset L'$;
  when the value of a type (calculated as a Boolean expression) is *true*, the transition can become active, otherwise it cannot.

The following ordered four-tuple

$$E = \langle \langle A, \pi_A, \pi_L, c, f, \Theta_1, \Theta_2 \rangle, \langle K, \pi_k, \Theta_K \rangle, \langle T, t^0, t^* \rangle, \langle X, \Phi, b \rangle \rangle \qquad (2)$$

is called *generalized net* if the elements are described as follows:

- A is a set of transitions,
- π_A is a function giving the priorities of the transitions, i.e. $\pi_A : A \to N$, where $N = \{0, 1, 2, \ldots\} \cup \{\infty\}$,
- π_L is a function giving the priorities of the places, i.e. $\pi_L : L \to N$, where $L = pr_1 A \cup pr_2 A$, and $pr_i X$ is the i-th projection of the n-dimensional set, where $n \in N$, $n \ge 1$ and $1 \le i \le n$ (obviously, L is the set of all generalized nets places),
- c is a function giving the capacities of the places, i.e. $c : L \to N$,
- f is a function that calculates the truth values of the predicates of the transition's conditions (the function

f have the value *false* or *true*, i.e. a value from the set $\{0, 1\}$,

- Θ_1 is a function giving the next time-moment when a given transition Z can be activated, i.e. $\Theta_1(t) = t'$, where $pr_3 Z = t$, $t' \in [T, T + t^*]$ and $t \le t'$; the function value is calculated at the moment when the transition terminates its functioning,
- Θ_2 is a function giving the duration of the active state of a given transition Z, i.e. $\Theta_2(t) = t'$, where $pr_4 Z = t \in [T, T + t^*]$ and $t' \ge 0$; the value of this function is calculated at the moment when the transition starts its functioning,
- K is the set of the generalized net's tokens; in some cases, it is convenient to consider it as a set of the form

$$K = \bigcup_{l \in Q^I} K_l$$

where K_l is the set of tokens that enter the net from place l, and Q^I is the set of all input places of the net,
- π_K is a function giving the priorities of the tokens, i.e. $\pi_K : K \to N$,
- Θ_K is a function giving the time-moment when a given token can enter the net, i.e. $\Theta_K(\alpha) = t$, where $\alpha \in K$ and $t \in [T, T + t^*]$,
- T is the time-moment when the generalized net starts functioning; this moment is determined with respect to a fixed (global) time-scale,
- t^0 is an elementary time-step, related to the fixed (global) time-scale,
- t^* is the duration of the generalized net functioning,
- X is the set of all initial characteristics the tokens can receive on entering the net,
- Φ is a characteristic function that assigns new characteristics to every token when it makes the transfer from an input to an output place of a given transition,
- b is a function giving the maximum number of characteristics a given token can receive, i.e. $b : K \to N$; for example, if $b(\alpha) = 1$ for some token α, then this token will enter the net with some initial characteristic and subsequently it will keep only its current characteristic; when $b(\alpha) = \infty$ the token α will keep all its characteristics; when $b(\alpha) = k < \infty$ the token α will keep its last k characteristics (the characteristics older than the last k will be forgotten); in the general case every token α has $b(\alpha) + 1$ characteristics on leaving the net.

A given generalized net may lack some of the above components. In these cases, any missing component will be omitted. The generalized nets of this kind form a special class and is called *reduced generalized nets*.

3. THE ALGORITHM OF GENERALIZED NETS

The algorithms for movement of tokens in the generalized net are more general than those of Petri nets. In a Petri net implementation, parallelism is reduced to a sequential firing of its transitions and the order of their activation in the general case is probabilistic or dependent on the transitions' priorities, if such exist. The generalized net's algorithms provide a means for a more detailed modelling of the described process. The algorithms for the token's transfers take into account the priorities of the places, transitions and tokens, i.e. e. they are more precise.

Following [1] we will describe the *general algorithm* for the transitions functioning which is equivalent to the tokens transfer algorithm.

The general algorithm starts at time $t_1 = TIME$ and has the following construction:

1. sort the input places of the transitions by their priorities (there are two groups of tokens, the first group (P_1) contains tokens which can be directly transform to the transition output places, and the second one (P_2) will be described in step 6),
2. sort the tokens from group (P_1) of the input places by their priorities, (let the index matrix *R* - related to the index matrix *r* – be defined as follows:

$$R_{ij} = \begin{cases} 1 & \text{if the} (i, j) - \text{th predicate } r_{ij} \text{ is } \textit{true} \\ 0 & \text{if the} (i, j) - \text{th predicate } r_{ij} \text{ is } \textit{false} \text{ or} \\ & \text{if the value is determined in step 3} \end{cases}$$

3. assign a value *0* to all elements of *R* for which either
   - the input place which corresponds to the respective predicate is empty (the group P_1 is empty), or
   - the output place which corresponds to the respective predicate is full, or
   - the current capacity of the arc between the corresponding input and output places is *0*,
4. calculate the values of the other elements of *r* and assign these values to the elements of *R*,
5. calculate the values of the characteristic functions related to the corresponding output places in which tokens will enter, and assign the current characteristics of the entering tokens,
6. for each input place (considering the input place priorities) do as follows:

– select the tokens with the highest priority in the input place,
– transfer the selected tokens to all output places for which the corresponding predicate enables this (these tokens constitute the group P_2 of the output places),

7. transfer the tokens with the highest priority for which all calculated values of the predicates are equal to *false* to the group P_2 of the corresponding places; in this group transfer also all tokens which cannot be transferred to the corresponding output places because these places are already full with tokens from other places of higher priorities,
8. increase the time, $TIME := TIME + t^0$,
9. check whether the value of the current time is smaller than $t_1 + t_2$,
10. if the answer (in step 9) is *yes* go to step 2,
11. if the answer (in step 9) is *no* terminate the current functioning of the token.

4. ALGEBRAIC APPROACH TO GENERALIZED NETS

For two given transitions Z_1 and Z_2 we will define

$$Z_1 = Z_2 \ \underline{\text{iff}} \ (\forall i : 1 \le i \le 7)(pr_i Z_1 = pr_i Z_2);$$

$$\begin{aligned} Z_1 \subset Z_2 \ \text{iff} \ & (\forall i : 1 \le i \le 2)(pr_i Z_1 \subset pr_i Z_2) \\ & \& (\forall i : 3 \le i \le 4)(pr_i Z_1 = pr_i Z_2) \\ & \& (\forall i : 5 \le i \le 6)(pr_i Z_1 \subset pr_i Z_2) \\ & \& (pr_7 Z_1 \subset_2 pr_7 Z_2), \end{aligned}$$

where:

- $\subset_1$ is a relation of inclusion over index matrices and if $A = [K_1, L_1, \{a_{i,j}\}]$, $B = [K_2, L_2, \{b_{i,j}\}]$, , then
$$A \subset_1 B \ \underline{\text{iff}} \ \begin{array}{l} (K_1 \subset K_2) \ \& \ (L_1 \subset L_2) \ \& \\ (\forall i \in K_1)(\forall j \in L_1)(a_{i,j} = b_{i,j}) \end{array};$$
- $\subset_2$ is a relation of inclusion over Boolean expressions and, for two such expressions a and b <u>iff</u> the expression a is obtained after removing a part of the arguments of b and the logical operations associated to them.

We will define four operations over the transitions

$$Z_i = < L_1^i, L_2^i, t_1^i, t_2^i, r^i, M^i, \square^i, > (i = 1, 2)$$

The following statement must necessarily hold: if place $l \in pr_1 Z_i \cap pr_2 Z_i$ and $l \in pr_s Z_{3-i}$, then $l \in pr_{3-s} Z_{3-i}$ for $1 \le i \le 2$, $1 \le s \le 2$. These operations are

a) a *union* (the necessary conditions for this operation are $t_j^1 = t_j^2 \ (j = 1, 2)$, and if $l \in pr_s Z_i$, and it is not allowed that $l \in pr_{3-s} Z_{3-i}$ for $1 \le i \le 2$, $1 \le s \le 2$:
$$\begin{array}{c} Z_1 \cup Z_2 = \langle L_1^1 \cup L_1^2, L_2^1 \cup L_2^2, t_1^1, t_2^1, r^1 + r^2, \\ M^1 + M^2, \vee(\square^1, \square^2) \rangle \end{array};$$

b) an *intersection* (with the above conditions):
$$\begin{array}{c} Z_1 \cap Z_2 = \langle L_1^1 \cap L_1^2, L_2^1 \cap L_2^2, t_1^1, t_2^1, r^1 \times r^2, \\ M^1 \times M^2, \wedge(\square^1, \square^2) \rangle \end{array};$$

c) a *composition* (with the above condition and with the condition $L_1^1 \cap L_1^2 = L_2^1 \cap L_2^2 = \varnothing$):
$$\begin{array}{c} Z_1 o Z_2 = \langle L_1^1 \cup (L_1^2 - L_2^1), \ L_2^2 \cap (L_2^1 - L_1^2), t_1^1, t_2^1 + t_2^2, \\ r^1.r^2, M^1.M^2, \vee(\square^1, \overline{\square}^2) \rangle \end{array},$$
where $\overline{\square}$ can be obtained from $\square$ after removing all its arguments whose identifiers are elements of the set $L_2^1 \cup L_1^2$.
It is possible that $L_1^1 \cap L_1^2 = \theta$ and/or $L_2^1 \cap L_2^2 = \theta$. In this case $Z_1 \cap Z_2 Z_\varnothing$ is the empty transition, i.e., a transition without places (some other components of it such as $M, r, \square$ will also be degenerated).

d) A *difference* (with the above conditions):
$$Z_1 - Z_2 = \begin{cases} Z_\varnothing & \text{if } L_1^1 \subset L_1^2 \text{ or } L_1^2 \subset L_2^2 \\ Z_1 - Z_2 = & \langle L_1^1 - L_1^2, L_2^1 - L_2^2, t_1^1, t_2^1, r^1 - r^2, \\ & M^1 - M^2, \square^1 / \square^2 \rangle, \text{ otherwise} \end{cases}$$

where $\square^1 / \square^2$ results from $\square^1$ after removing all its arguments whose identifiers are element of the set $L_1^1 \cap L_2^1$.

The operations described below do not exist elsewhere in the Petri net theory. The can be transferred to practically all other types of Petri nets. These operations are useful for constructing generalized net models of real processes.

Before introducing the different generalized net operations, we will formulate some appropriateness conditions for the arguments of these operations.

We will assume that if a place participates simultaneously in two generalized nets, then in both of them it has:

a) equal capacities,
b) the same characteristic functions,
c) equal possibility to host tokens from K, that is, if in the first net (which is a model of some process) the tokens which are elements of $K' \subset K$ pass through the place and in the second net tokens from the set $K'' \subset K$ can also pass through this place, then $K' = K''$,
d) equal values of the priority functions, and
e) the same number (identification).

Similarly, we will expect that if two places are respectively input and an output one for two transitions in both nets, then

a) the capacities of the connecting arcs are equal,
b) the time for passing from the input to the output place is the same,
c) the place which is an input (output) one in one of the transitions should play the same role in the other transition.

Let E_1 and E_2 be two generalized nets and let for $1 \le i \le 2$:

$$E_i = \langle\langle A_i,\ \theta_1^i \pi_L^i, c^i,\ f^i, \theta_1^i,\ \theta_2^i\rangle, \langle K_i, \pi_K^i\ \theta_K^i\ \rangle,$$
$$\langle T_i, t_i^o, t_i^*\rangle, \langle X_i, \Phi_i, b_i\rangle\rangle$$

A *union* will be called the object:

$$E_1 \cup E_2 = \langle\langle A_1 \,\overline{\cup}\, A_2, \pi_A^1 \cup \pi_A^2, \pi_L^1 \cup \pi_L^2, c^1 \cup c^2,$$
$$f^1 \cup f^2, \theta_1^1 \cup \theta_1^2,$$
$$\theta_2^1 \cup \theta_2^2\rangle, \langle K_1 \cup K_2, \pi_K^1 \cup \pi_K^2, \theta_K^1 \cup \theta_K^2\rangle,$$
$$\langle \min(T_1,T_2), GCD(t_1^o,t_2^o), \max_{1\le i\le 2}\left(T_i + \frac{t_i^* \cdot t_i^o}{GCD(t_1^o,t_2^o)} - \right.$$
$$\left. - \min(T_1,T_2)\right)\rangle, \langle X_1 \cup X_2, \Phi_1 \cup \Phi_2, b_1 \cup b_2\rangle\rangle$$

where

$$A_1 \,\overline{\cup}\, A_2 = \bigcup_{i=1}^{2}\{Z \mid (Z \in A_i)\ \&\ (\forall Z' \in A_{3-i})(Z \cap Z' = Z_\theta)\} \cup$$
$$\bigcup_{i=1}^{2}\{Z \mid (\exists Z' \in A_i)(\exists Z'' \in A_{3-i})(Z' \cap Z'' \ne Z_\theta)(Z = Z' \cup Z'')\}.$$

A *composition* of the above nets will be called the object:

$$E_1 o E_2 = \begin{cases} E_1, & \text{if } T_2 + t_2^* < T_1 \\ E_3, & \text{if } T_1 \le T_2 + t_2^* \end{cases},$$

where

$$E_3 = \langle\langle A_1 \,\overline{\cup}\, A_2, \pi_A^1 \cup \pi_A^2, \pi_L^1 \cup \pi_L^2, c^1 \cup c^2,$$
$$f^1 \cup f^2, \theta_1^1 \cup \theta_1^2, \theta_2^1 \cup \theta_2^2\rangle$$
$$\langle K_1 \cup K_2, \pi_K^1 \cup \pi_K^2, \theta_K^1 \cup \theta_K^2\rangle,$$
$$\langle (T_1, GCD(t_1^o,t_2^o), \max_{1\le i\le 2}(T_i + \frac{t_i^* \cdot t_i^o}{GCD(t_1^o,t_2^o)} - T_1)\rangle,$$
$$\langle X_1 \cup X_2, \Phi_1 \cup \Phi_2, b_1 \cup b_2\rangle\rangle$$

A *difference* of the two nets will be called the object:

$$E_1 - E_2 = \langle\langle A_1 -_* A_2, \pi_A^1 -_* \pi_A^2, \pi_L^1 -_* \pi_L^2,$$
$$c^1 -_* c^2,\ f^1 -_* f^2,$$
$$\theta_1^1 -_* \theta_1^2, \theta_2^1 -_* \theta_2^2\rangle, \langle K_1 - K_2, \pi_K^1 -_* \pi_K^2, \theta_K^1 -_* \theta_K^2\rangle,$$
$$\langle T_1, t_1^o, t_1^*\rangle, \langle X_1 - X_2, \Phi_1 -_* \Phi_2, b_1\rangle\rangle,$$

where

$$A_1 -_* A_2 = \{Z \mid (Z \in A_1)\ \&\ (\forall Z' \in A_2)(Z \cap Z', = Z_\theta)\} \cup$$
$$\{Z \mid (\exists Z' \in A_1)(\exists Z'' \in A_2)(Z' \cap Z'' \ne Z_\theta)\ \&\ (Z = Z' - Z'')\},$$

$\pi_A^1 -_* \pi_A^2$ is obtained from π_A^1 after removing all its arguments whose identifiers are not elements of the set $A_1 -_* A_2$; $\pi_L^1 -_* \pi_L^2$ is obtained from π_L^1 after removing all its arguments whose identifiers are not elements of the set $L_1^1 - L_2^1$, etc.

Let us define for a given generalized net E two sets, K of all tokens and X of all initial characteristics:

$$K(E) = \{\alpha \mid (\forall \alpha \in K)(\theta_K(\alpha) < T + t^*)\},$$
$$X(E) = \{x_0^\alpha \mid (\forall \alpha \in K(E))(x_0^\alpha \in X)\},$$

and let $X(\alpha)$ be the set of all different characteristics the token α can have initially. Obviously,

$$X(E) \subset \bigcup_{\alpha \in K} X(\alpha).$$

For a given token $\alpha \in K$ and a given initial characteristic $x \in X(E)$ let

$$E(\alpha, x) = \begin{cases} <\alpha, x_{f_{in}}^\alpha>, & \text{if } \alpha \in K(E) \text{ and } x \in X(\alpha) \\ <\alpha, x>, & \text{otherwise} \end{cases},$$

$$E(\alpha, x) = \begin{cases} <\alpha, x, x_1^\alpha, x_2^\alpha, \ldots, x_{f_{in}}^\alpha>, & \text{if } \alpha \in K(E) \text{ and } x \in X(\alpha) \\ <\alpha, x>, & \text{otherwise} \end{cases},$$

where $x_{f_{in}}^\alpha$ is the final characteristic of the token α in the generalized net E and $x_1^\alpha, x_2^\alpha, \ldots, x_{f_{in-1}}^\alpha$ are the rest of characteristics the token has received during its transfer in the net.

The first generalized net definition is in some sense deductive. Below we will introduce a second definition of a generalized net. In the above sense, the new definition will be of inductive nature.

Let Z be a given object, having the graphical structure shown in Figure 2 and the following components:

a) L' – a set of places called input places;
b) L'' – a set of places called output places;
c) t_1 – a time-moment taken with respect to some fixed time-scale (with an elementary time-step t^o);
d) t_2 – a real number which corresponds to the length of the of the time-interval in the above mentioned time-scale;
e) r – an index matrix having the from

$$r = \begin{array}{c|c} & l''_1 \ldots\ l''_j \ldots\ l''_n \\ \hline l'_1 & \\ \vdots & r_{i,j} \\ l'_i & (r_{i,j} - \text{predicate}) \\ \vdots & (1 \le i \le m, 1 \le j \le n) \\ l'_m & \end{array}$$

(i, j) -th element of which is a predicate and corresponds to the i -th input and j -th output place;

f) M , and index matrix having the form

$$r = \begin{array}{c|c} & l''_1 \ldots l''_j \ldots l''_n \\ \hline l'_1 & \\ \vdots & m_{i,j} \\ l'_i & (m_{i,j} \geq 0 - \text{natural number}) \\ \vdots & (1 \leq i \leq m, 1 \leq j \leq n) \\ l'_m & \end{array}$$

g) □ is an object having a from similar to a Boolean expression. Its variables are exactly the names of Z's input places.

The object described above will be called a *transition* .

The inductive definition of the concept of generalized net is as follows:

1. An object that has the from of a transition is called a generalized net if the following are added to it:
   a) π_A – a function giving a natural number (transition priority);
   b) π_L – a function giving the priorities of the transition's places;
   c) c – a function giving the capacities of the transition's places;
   d) f – a function which calculates the truth values of the predicates of the index matrix r ;
   e) θ_1 – a function giving the next time-moment when the given transition can be activated, i.e., $\theta_1(t) = t'$, where $t, t' \in [T, T + t^*]$ and $t \leq t'$. The value of this function is calculated at the moment when the transition terminates its functioning;
   f) θ_2 – a function giving the duration of the active state of a given transition, i.e., $\theta_2(t) = t'$, where $t \in [T, T + t^*]$ and $t' \geq 0$. The value of this function is calculated at the moment when the transition starts its functioning;
   g) K – a set of tokens;
   h) π_K – a function giving the priorities of the tokens;
   i) θ_K – a function giving the time-moment when a given token can enter the net, i.e., $\theta_K(\alpha) = t$, where $\alpha \in K, t \in [T, T + t^*]$;
   j) T – a time-moment when generalized net starts functioning. This moment is determined with respect to a fixed time-scale and the first value of $t_1 = T$;
   k) t^o – an elementary time-step related to the fixed time-scale;
   l) t^* – duration of the net functioning;
   m) X – a set of all initial characteristics witch the token can receive on entering the net;
   n) Φ – a characteristic function which assigns new characteristics to every token when it makes a transfer from an input to an output place of any transition;
   o) b – a function giving the maximum number of characteristics, which a given token can receive during its transfer in the net.
2. If E_1 and E_2 are generalized nets, then $E_1 \cup E_2$ is a generalized net.

5. OPERATOR ASPECTS OF GENERALIZED NETS

Operations and relations are defined as over the transitions, as well as over the generalized nets in general.

The operations, defined over the generalized nets – union", "intersection"', "`composition"' and "iteration" do not exist anywhere else in the Petri net theory. They can be transferred to virtually all other types of Petri nets (obviously with some modifications concerning the structure of the corresponding nets). These operations are useful for constructing generalized net models of real processes.

Now, the operator aspect has an important place in the theory of generalized nets. Six types of operators are defined in its framework. Every operator assigns to a given generalized net a new generalized net with some desired properties. The comprised groups of operators are:

- global (*G*-) operators,
- local (*P*-) operators,
- hierarchical (*H*-) operators,
- reducing (*R*-) operators,
- extending (*O*-) operators,
- dynamic (*D*-) operators.

The *global operators* transform, according to a definite procedure, a whole given net or all its components of a given type. There are operators that alter the form and structure of the transitions (G_1, G_2, G_3, G_4, G_6) temporal components of the net (G_7, G_8); the duration of its functioning (G_9), the set of tokens (G_{10}), the set of the initial characteristics (G_{11}); the characteristic function of the net (G_{12}) (this function is the union of all places' characteristic functions); the evaluation function (G_{13}), or other net's functions (G_5, G_{14}, G_{15}, G_{16}, G_{17}, G_{18}, G_{19}, G_{20}).

One of the global operators can collapse a given generalized net to a generalized net-transition (G_2). Another operator (G_4) adds two special places and, connected to two special transitions of the generalized net: a *general input place*, where all tokens enter the net and are later distributed among the net's actual input places; and a *general output place* that collects all tokens leaving the generalized net from their respective output places.

Another global operator (G_3) transforms a given generalized net after its functioning so that it removes all tokens which have not participated in the process and all places which have not been visited by tokens. The new net has the same functional behaviour as the original one; however, all its tokens and places are actually involved in the modelled process.

Some global operators (G_5, G_{12}, G_{13}, G_{14}, G_{15}, G_{16}, G_{17}, G_{18}, G_{19}, G_{20}) alter the different (global) functions defined on the net.

The second types of operators are *local operators*. They transform single components of some of the transitions of a given generalized net. There are three types of local operators:

- temporal (P_1, P_2, P_3, P_4), that change the temporal components of a given transition,
- matrix (P_5, P_6,), that change some of the index matrices of a given transition,
- other operators: these alter the transition's type (P_7), the capacity of some of the places in the net (P_8) or the characteristic function of an output place (P_9), or the evaluation function associated with the transition condition predicates of the given transition (P_{10}).

For any of these operators, a continuation (P_i, $1 \le i \le 10$), to a global one ($\overline{P}_i, 1 \le i \le 10$) can be made by defining the corresponding operator in such a way that it would transform all components of a specified type in every transition of the net.

The third types of operators are the *hierarchical operators*. These are of five different types and fall into two groups according to their way of action:

- expanding a given generalized net (H_1, H_2 and H_5),
- shrinking a given generalized net (H_2, H_4 and H_5).

The operator H_5 can be expanding as well as shrinking, depending on its form.

According to their object of action the operators fall again into two groups:

- acting upon or giving as a result a place (H_1 and H_2),
- acting upon or giving as a result a transition (H_3, H_4 and H_5).

The hierarchical operators H_1 and H_3 replace a given place or transition, respectively, of a given generalized net with a whole new generalized net. Conversely, operators H_2 and H_4 replace a part of a given generalized net with a single place (H_2) or transition (H_4). Finally, the operator H_5 changes a subnet of a given generalized net with another subnet. Expanding operators can be viewed as tools for magnifying the modelled process' structure; while shrinking operators - as a means of integration and ignoring the irrelevant details of the process.

The next (fourth) group of operators defined over the generalized nets produce a new, reduced generalized net from a given net. They would allow the construction of elements of the classes of reduced generalized nets. To find the place of a given Petri net modification among the classes of reduced generalized nets, it must be compared to some reduced generalized net obtained by an operator of this type. These operators are called *reducing operators*.

Operators from the fifth group extend a given generalized net. These operators are called *extending operators*. The extending operators are associated with every one of the generalized net extensions.

Finally, the operators from the last - sixth - group are related to the ways the generalized net functions, so that they are called dynamic operators. These are the following:

- operators $D(1, i)$ that determine the procedure of evaluating the transition condition predicates $(1 \le i \le 18)$,
- operators governing token splitting: one that allows $D(2, 1)$ and one that prohibits splitting $D(2, 2)$, respectively; and operators governing the union of tokens having a common predecessor: an allowing one $D(2, 4)$ and a prohibiting one $D(2, 3)$,
- operators that determine the strategies of the tokens transfer: one by one at a time vs. all in groups (the operator $D(3, 2)$; the operator $D(3, 1)$ does not allow this),
- operators related to the ways of evaluating the transition condition predicates: predicate checking $D(4, 1)$; changing the predicates by probability

functions with corresponding forms $D(4, 2)$; expert estimations of predicate values $D(4, 3)$; predicates depending on solutions of optimisation problems (e.g. transportation problem) $D(4, 4)$.

The operators of different types, as well as the others that can be defined have a major theoretical and practical value. On the one hand, they help us study the properties and the behaviour of generalized nets. On the other hand, they facilitate the modelling of many real processes. *A Self-Modifying generalized net* can be constructed and described, the Self-Modifying generalized net has the property of being able to alter its structure (number of transitions, places, tokens, transition condition predicates, token characteristics, place and arc capacities, etc.) and the token transfer strategy during the time of the generalized net-functioning. These changes are done by operators that can be defined over the generalized net. Of course, not all operators can be applied over a generalized net during the time of its functioning. Some of them are only applicable before, and others - after the generalized net-functioning.

The necessary conditions for an operator to be applicable during the time of the generalized net-functioning are discussed in [1].

6. APPLICATIONS

The concept of generalized nets methodology was described for modeling discrete event systems as well as the concept of index matrix used in this kind of modeling.

Generalized nets can be used for modeling any kind of systems [2] and [3] in order to represent knowledge of considered systems, like: - ecological systems, - economic systems, - biological systems, etc. with different types of used methodology, e.g. fuzzy logic, rough sets, neural networks and so on.

As an example we recall a neural network functioning model [5] and [6] shown in Figure 2 and 3. The models consider different kind of aggregation of information within a neural network.

In this paper we showed only an overview of the generalized nets modeling.

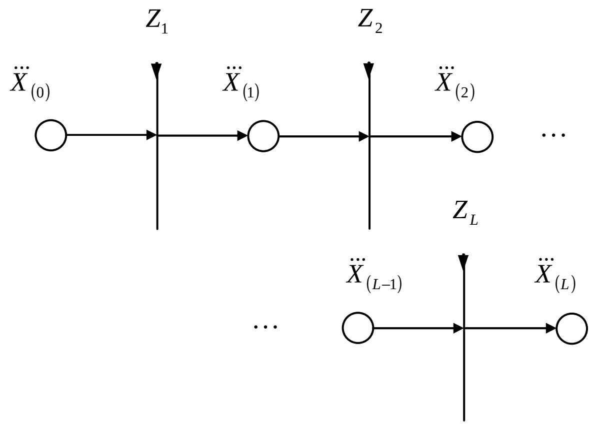

Figure 2. The Generalized Net model of neural network with layer aggregation.

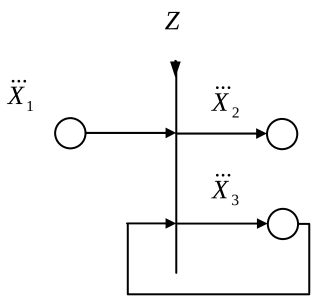

Figure 3. The Generalized Net model of the aggregated neural network.

REFERENCES

[1]. Atanassov, K., (1991) *Generalized nets*. World Scientific, Singapore, New Jersey, London.

[2]. Atanassov, K., (1997) *Generalized Nets and Systems Theory*. „Prof. M. Drinov" Academic Publishing House, Sofia.

[3]. Radeva, V., Krawczak, M., Choy, E. (2002) Review and Bibliography on Generalized Nets Theory and Applications. *Advanced Studies in Contemporary Mathematics*, 4, 2, 173-199.

[4]. Atanassov, K., (1987) Generalized Index Matrices. *Competes Rendus de l'Academie Bulgare des Sciences*, 40, 11, 15-18.

[5]. Krawczak, M. (2003) *Multilayer Neural Systems and Generalized Net Models*. Academic Press EXIT, Warsaw, Poland.

[6]. Krawczak, M. (2004) Generalized Nets Modeling Concept: Neural Networks Models. In: Grzegorzewski P., Krawczak M., Zadrozny S. (Eds.): *Soft Computing. Tools, Techniques and Applications*. Akademicka Oficyna Wydawnicza EXIT, Warszawa 2004.