BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **10**, No 4

Sofia • 2010

Algorithms for Tokens Transfer in Different Types of Intuitionistic Fuzzy Generalized Nets

Krassimir Atanassov[1], *Dimitar Dimitrov*[1], *Vassia Atanassova*[2]

[1] *Institute of Biophysics and Biomedical Engineering, 1113 Sofia*
[2] *Institute of Information and Communication Technologies, 1113 Sofia*
E-mails: krat@bas.bg   mitex.gbg.bg   vassia.atanassova@gmail.com

Abstract: *The paper presents formal definitions of the concepts Intuitionistic Fuzzy Generalized Nets from first, second, third and fourth type and describes the algorithms for tokens transfer within the separate transitions and the nets in general.*

Keywords: *Algorithm, Generalized net, Intuitionistic fuzzy estimation.*

1. Introduction

Generalized Nets (GNs, see [4, 6]).) are extensions of Petri nets (see, e.g. [9]) and Petri net modifications and extensions.

Since 1983 more than 800 papers, being related to the concept of the GNs have been published. A part of them are included in the bibliographies [1, 8].

GNs have so far over 25 extensions [4, 6]. For each one of them it is proved that it is a conservative extension, i.e., its functioning and results of the work can be described by an ordinary GN.

Intuitionistic Fuzzy Sets (IFSs), defined in 1983, are extensions of fuzzy sets (see [5]). They have two degrees: a degree of membership (μ) and a degree of non-membership (ν), such that their sum may be smaller than 1, i.e., a third degree of uncertainty ($\pi = 1 - \mu - \nu$) can be defined, too. A variety of operations, relations and operators (from modal, topological and others types) are defined over IFSs.

The first two extensions of the GN, proposed in 1985 (see [2, 4, 6]), were called Intuitionistic Fuzzy GNs (IFGNs) of first and second types. Two other extensions are described in [7] in 2001 (see also [6]). In the present paper detailed algorithms for IFGN-functioning for each of the four types of nets will be given. These algorithms are more effective than the present ones, described in [4].

2. Basic elements of Intuitionistic Fuzzy Logic (IFL)

To each proposition (in the classical sense) we can assign its truth value: truth – denoted by 1, or falsity – 0. In the case of fuzzy logic this truth value is a real number in the interval [0, 1] and may be called "truth degree" of a particular proposition. Here we add one more value – "falsity degree" – which will be in the interval [0, 1] as well. Thus two real numbers, $\mu(p)$ and $\nu(p)$, are assigned to the proposition p with the following constraint to hold (see [5]):

$$\mu(p) + \nu(p) \leq 1.$$

The degree of uncertainty (indeterminacy) is defined as

$$\pi(p) = 1 - \mu(p) - \nu(p).$$

Let this assignment be provided by an evaluation function V defined over a set of propositions S in such a way that:

$$V(p) = \langle \mu(p), \nu(p) \rangle.$$

For the needs of the discussion below we shall define the notions of

- intuitionistic fuzzy tautology (IFT) through:

  "A is an IFT" if and only if $V(A) = \langle a, b \rangle$, then $a \geq b$.

- intuitionistic fuzzy sure (IFS) through:

  "A is an IFS" if and only if $V(A) = \langle a, b \rangle$, then $a \geq \frac{1}{2}$.

3. Intuitionistic Fuzzy Generalized Nets of first type (IFGN1s)

GNs are defined as extensions of the ordinary Petri nets and their modifications, but in a way that is principally different from the ways of defining the other types of Petri nets. The additional components in the GN-definition provide more and greater modeling possibilities and determine the place of GNs among the individual types of Petri nets, similar to the place of the Turing machine among finite automata.

The IFGN1s is the first extension of the GN. It keeps all GN-components, but some of them are modified.

The first basic difference between IFGNs (and of GNs, in general) and the ordinary Petri nets is the "place – transition" relation [9]. Here, the transitions are objects of a more complex nature. An IFGN1-transition contains m input and n output places where $m, n \geq 1$.

Formally, every IFGN1-transition (and GN-transition) is described by a seven-tuple Z as illustrated on Fig. 1.

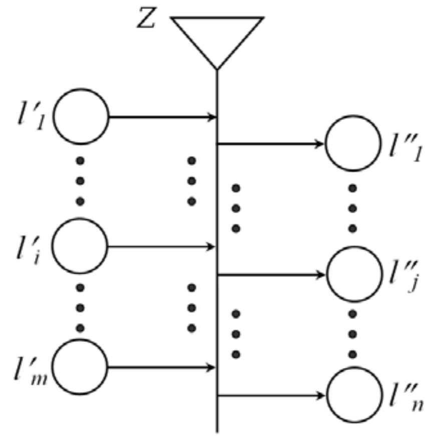

Fig. 1

$$Z = \langle L', L'', t_1, t_2, r, M, \square \rangle$$

where:

(a) L' and L'' are finite, non-empty sets of places (the transition's input and output places, respectively); for the transition in Fig. 1 these are $L' = \{l'_1, l'_2, ..., l'_m\}$ and $L'' = \{l''_1, l''_2, ..., l''_n\}$;

(b) t_1 is the current time-moment of the transition's firing;

(c) t_2 is the current value of the duration of its active state;

(d) 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 (IM; see [3]):

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

$r_{i,j}$ is the predicate that corresponds to the i-th input and j-th output places. When its intuitionistic fuzzy estimation satisfies at least one of the seven conditions given by point **(A'05)** in Section 4, a token from i-th input place transfers to j-th output place; otherwise, this is not possible;

(e) M is an index matrix of the capacities of transition's arcs:

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

(f) □ 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 is an expression built up from variables and the Boolean connectives $\wedge$ and $\vee$ whose semantics is defined as follows

$\wedge(l_{i1}, l_{i2}, ..., l_{iu})$ – every place $l_{i1}, l_{i2}, ..., l_{iu}$ must contain at least one token;

$\vee(l_{i1}, l_{i2}, ..., l_{iu})$ – there must be at least one token in any of the places $l_{i1}, l_{i2}, ..., l_{iu}$ where $\{l_{i1}, l_{i2}, ..., l_{iu}\} \subset L'$

When the value of a type (calculated as a Boolean expression) is "true", the transition can become active, otherwise it cannot.

The ordered four-tuple

$$E = \langle\langle A, \pi_A, \pi_L, c, f, \theta_1, \theta_2\rangle, \langle K, \pi_K, \theta_K\rangle, \langle T, t^\circ, t^*\rangle, \langle X, \Phi, b\rangle\rangle$$

is called an IFGN1 if:

(a) A is a set of transitions;

(b) π_A is a function giving the priorities of the transitions, i.e., $\pi_A : A \to N$, where $N = \{0, 1, 2, ...\} \cup \{\infty\}$;

(c) π_L is a function giving the priorities of the places, i.e., $\pi_L : L \to N$, where $L = pr_1A \cup pr_2A$, and pr_iX is the i-th projection of the n-dimensional set, where $n \in N$, $n \geq 1$ and $1 \leq k \leq n$ (obviously, L is the set of all GN-places);

(d) c is a function giving the capacities of the places, i.e., $c : L \to N$;

(e) f is a function that calculates the truth values of the predicates of the transition's conditions. For an ordinary GN, function f has values from the set $\{0, 1\}$, while for IFGN1 its values have the form $\langle \mu_{i,j}, \nu_{i,j}\rangle$, where $\mu_{i,j}$ and $\nu_{i,j}$ are the degrees of validity and of non-validity of the predicate staying on place (i, j) in the index matrix and $\mu_{i,j} + \nu_{i,j} \leq 1$;

(f) 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 $pr_3Z = t$; $t' \in [T, T + t^*]$ and $t \leq t'$. The value of this function is calculated at the moment when the transition terminates its functioning;

(g) θ_2 is a function giving the duration of the active state of a given transition Z, i.e., $\theta_2(t) = t'$, where $pr_4Z = t$; $t' \in [T, T + t^*]$ and $t' \geq 0$. The value of this function is calculated at the moment when the transition starts its functioning;

(h) K is the set of the GN's tokens;

(i) π_K is a function giving the priorities of the tokens, i.e., $\pi_K : K \to N$;

(j) 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 $t \in [T, T + t^*]$;

(k) T is the time-moment when the GN starts functioning. This moment is determined with respect to a fixed (global) time-scale;

(l) $t°$ is an elementary time-step, related to the fixed (global) time-scale;

(m) t^* is the duration of the GN functioning;

(n) X is the set of all initial characteristics the tokens can receive on entering the net.

(o) Φ is a characteristic function that in the case of an ordinary GN assigns new characteristics to every token when it makes the transfer from an input to an output place of a given transition, while in the case of an IFGN1 adds to the same characteristic the degrees of validity and of the non-validity of the predicate that allows the token's transfer. So, when some token finishes with its transfer in the IFGN1 we can determine the degrees of validity and of the non-validity with which it realized its transfer.

(p) b is a function giving the maximum number of characteristics a given token can receive, i.e., $b : K \to N$.

A given IFGN1 may lack some of the above components. In these cases, any missing component will be omitted. The GNs of this kind form a special class of GNs called "reduced IFGN1s".

All operations and relations that are defined over GNs, can be defined also on IFGN1s. The operations, defined over the GNs – "union", "intersection", "composition" and "iteration" (see [4, 6]) 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 GN models of real processes.

4. Algorithms for transition and IFGN1 functioning

The IFGN1 definition is more complex than the definition of a Petri net and of the other Petri net modifications, as well as than the definition of the ordinary GNs. In a Petri net implementation, parallelism is reduced to a sequential firing of the net transitions and in general the order of their activation is probabilistic or dependent on the transitions' priorities, if ones exist. The GN's algorithms enable a more detailed modeling of the described process. The algorithms for the token's transfers take into account the priorities of the places, transitions and tokens, i.e., they are more precise.

In [6], a more detailed algorithm for an ordinary GN-tokens transfer than in [4] is given. Now, we will introduce for the first time a new and more detailed algorithm for IFGS1s, that can be transformed to the token transfer in the ordinary GNs.

By analogy with [4, 6], some components of the IFGN1's definition were not given above because they are related to the algorithm described below. They are explicitly mentioned in the text.

Following and modifying [6], we will propose the general algorithm for a transition's functioning. After this we will describe the procedure of the entire

IFGN1's functioning. The general algorithm (which we will denote by **Algorithm A'**) for tokens transfer after the time moment t_1 = TIME (the current IFGN1 time-moment) is as follows.

Algorithm A'

Step (A'01). The non-empty input places and the non-full output places are sorted by priority in descending order. An important addition to the IFGN1 transition description above, which is related to the software implementation of the transition's functioning, is the following. The tokens from a given input place are divided into two groups. The first one contains those tokens that can be transferred to the transition output, the second contains the rest (the motivation for this will be clear from the next steps of the algorithm). In the beginning, the second group is empty. Let the two groups be denoted by "$P_1(l)$" and "$P_2(l)$", respectively, where l is the corresponding place.

Step (A'02). Sort the tokens from P_1-groups of the input places (following the order from Step (A'01)) by their priorities.

Step (A'03). An empty index matrix R, which corresponds to the index matrix of the predicates r of the given transition, is generated. It is initiated without values. After this, we put values "$\langle 0,1\rangle$" (corresponding to value "false") of all of the elements of this index matrix, which:

– are placed in a row, corresponding to an empty input place, or
– are placed in a column, corresponding to a full output place, or
– are placed in (i, j)-th position, for which $m_{i,j} = 0$, i.e. the current capacity of the arc between i-th input and j-th output place is zero.

Step (A'04). The sorted places are passed sequentially by their priority, starting with the place having the highest priority, which has at least one token and through which no transfer has occurred on the current time-step. For the highest priority token (from the first list) we determine whether it can split or not. This fact is determined by the dynamical operator defined over the given GN and, particularly, over a given IFGN1 (see [6]). If such an operator is not defined, we will assume that a token can split as many times as necessary. After this the predicates corresponding to the relevant row of the index matrix R are checked. If the token cannot split, the check finishes with finding the first predicate with truth-value different from "$\langle 0, 1\rangle$"; in the other case, we must calculate the truth values of all predicates in the row, for which the elements of R are not zeros.

Step (A'05). Depending on the execution of the operator for permission or prohibition of tokens splitting over the net, the token from Step (A'04) will pass either to all permitted to it output places, or to this very place among them, which has the highest priority, following one of the seven conditions below, where for predicate $r_{i,j}$ it holds that $f(r_{i,j}) = \langle \mu_{i,j}, \nu_{i,j}\rangle$:

C1 $\mu_{i,j} = 1,\ \nu_{i,j} = 0$ (the case for an ordinary GN);

C2 $\mu_{i,j} > \frac{1}{2}\ (> \nu_{i,j})$;

C3 $\mu_{i,j} \geq \frac{1}{2}\ (\geq \nu_{i,j})$;

C4 $\mu_{i,j} > \nu_{i,j}$;

C5 $\mu_{i,j} \geq \nu_{i,j}$;

C6 $\mu_{i,j} > 0$;

C7 $\nu_{i,j} < 1$, i.e., at least $\pi_{i,j} > 0$.

The condition, that will be used, is determined for each transition before the IFGN1-firing. If one token cannot pass through a given transition on this time interval, it is moved to the second group of tokens of the corresponding input place. The tokens, which have entered into the place after the transition activation, are moved into the second list, too.

Step (A'06). The values of the characteristic function for the output places (one or more), in which tokens have entered (according to Step (A'05)), are calculated and the following record is assigned as a next token characteristic:

"value of function Φ for the respective token, $\mu_{i,j}, \nu_{i,j}$"

Step (A'07). Put values $\langle 0, 1\rangle$ in all rows of R for which the respective input place (from where the token goes out on Step (A'05)) is already empty; or in the columns of R which are full in a result of token's transfer on Step (A'05); or in the cells in the index matrix that correspond to arcs between the discussed input place and these output places, for which the arc-capacity becomes "0" as a result of the token's transfer.

Step (A'08). The current number of tokens in all input places for the current transition decrements with 1 for each token that has gone out of them at this time step. If the current number of tokens for a given input place is zero, the elements of the corresponding row of index matrix R are assigned the value "0".

Step (A'09). The capacities of all output places, in which a token, determined at Step (A'04), has entered, decrement with 1. If the maximum number of tokens for a given output place is reached, the elements of the corresponding column of the index matrix R are made "0".

Step (A'10). The capacities of all arcs, through which a token has passed, decrement with 1. If the capacity of an arc has reached 0, the element from the index matrix R that corresponds to this arc is assigned the value 0.

Step (A'11). If there are still tokens in the input places that are due to transfer, *and* there are spare output places, *and* there are arcs with non-zero capacities, then the algorithm proceeds to Step (A'12); in the opposite case it proceeds to Step (A'14).

Step (A'12). The current model time t is increased with t°.

Step (A'13). Is the current time moment equal to or greater than $t_1 + t_2$? If the answer to the question is "no", return to Step (A'04), otherwise go to Step **(A'14)**.

Step (A'14). Termination of the transition functioning.

Following [4, 6] we will propose the general algorithm for a GN or IFGN1's functioning (denoted by **Algorithm B' –** both coincide). For this purpose, we will introduce the concept of Abstract Transition (AT). This is a transition, which is the union of all active IFGN1-transitions at a given time moment. For its construction the operation "union" of transitions is used (see [6]). If we follow strictly the "union" operation's definition, we will end up with huge sparse matrices for AT's predicates and capacities. For faster software implementation, we will keep separate matrices for all transitions which form the AT, instead of merging them into single large matrices. In this way the number of outputs for a given input is significantly smaller than in classic ATs. Also removing of a transition from AT is easier.

Algorithm B'

Step (B'01). Put all tokens α for which $\theta_K(\alpha) \leq T$ into the corresponding input places of the net.

Step (B'02). Construct the IFGN1's AT (initially it is empty).

Step (B'03). Check whether the value of the current time is less than $T + t^*$.

Step (B'04). If the answer to the question on Step (B'03) is "no", terminate the IFGN1 process.

Step (B'05). Check all transitions for which the first time-component is exactly equal to the current time moment.

Step (B'06). Check the transition's Boolean types of all transitions determined by Step (B'05). The method of checking is as follows:

- change the names of all the places which participate in the Boolean expression of the transition type as variables with values: 0, if the corresponding place has no tokens at the current moment; 1, otherwise;
- calculate the truth value of the so obtained Boolean expression.

Step (B'07). If the transition's Boolean type is not satisfied, i.e., its truth value is 0, then function θ_1 calculates the subsequent moment of transition's firing, and the transition is removed from the AT. Otherwise, proceed to Step (B'08).

Step (B'08). Add all transitions from Step (B'06) for which the transition types are satisfied in the AT.

Step (B'09). Apply **Algorithm A'** over the AT for exactly one time step.

Step (B'10). Remove from AT all transitions which are inactive at the current time-moment.

Step (B'11). Increase the current time by t°.

Step (B'12). Go to Step (B'03).

5. Intuitionistic Fuzzy Generalized Nets of second type (IFGN2s)

In "Intuitionistic Fuzzy GNs of second type" (IFGN2, see [2, 4, 6]), the tokens are replaced by some "quantities of matter" that "flow" inside the net. This fact generates some differences between the definitions of IFGN1s and IFGN2s. Below we will show these differences.

The values of the transition condition's predicates can be intuitionistic fuzzy, that is, they can have degrees of truth and falsity as in IFGN1s. The IFGN2 has the form

$$E = \langle\langle A, \pi_A, \pi_L, c, f, \theta_1, \theta_2\rangle, \langle K, \pi_K, \theta_K\rangle, \langle T, t^\circ, t^*\rangle, \langle X, \Phi, b\rangle\rangle$$

where A is the set of the net's transitions (they have the ordinary GN form with only the following difference: here the capacities of the transition's arcs described in the index matrix M are real numbers).

The functions in the first component are similar to those in IFGN1 and they satisfy the same conditions, with the exception of function c that will assign a real number corresponding to the "volume" of the place, i.e., the quantity of matter that it can collect.

The essential difference between IFGN2s and the other GNs is the set K and the functions related to it. Now the elements of K are a kind of "quantities" which have as initial characteristics some "type" (elements of the set X) and which do not receive other characteristics. As in ordinary GNs, the function θ_K gives the time-moment when a token will enter the net.

The temporal components are the same as in the other types of GNs. Here, function Φ has a different meaning. Now it is related to the places, and by this function they receive characteristics (the quantities of the tokens from the different types in the corresponding places). As in GNs, it can be extended: it can also give other data for the modelled process (e.g., the time moments for the entry of the "quantities" in the places).

(d) c is a function giving the capacities of the places, i.e., $c : L \rightarrow R$, where R is the set of real numbers;

(h) K is the set of the GN's tokens that here represent quantities of matter;

(n) X is the set of all initial characteristics the tokens can receive on entering the net;

(o) Φ is a characteristic function that assigns a new characteristic to every place when some token makes a transfer from an input to it, and, as in the case of an IFGN1, adds to the same characteristic the degrees of validity and of non-validity of the predicate that allows the token's transfer;

(p) b is a function giving the maximum number of characteristics a given place can receive, i.e., $b : K \rightarrow N$.

A given IFGN2s, just like IFGN1, may have some of the above components missing. In these cases any missing component will be omitted from the net's record. The GNs of this kind form a special class of GNs called "*reduced IFGN2s*".

All operations and relations, which are defined over GNs, can be defined over IFGN2s, too.

6. Algorithms for transition and IFGN2 functioning

Now we will introduce for the first time the formal algorithm for IFGS2s. It will, of course, be based on the algorithms for token transfer in the ordinary GNs and in the IFGN1s, but now there are some essential differences and the first of them is that the tokens' transfer is done at one step only, yet with a duration determined by the second temporal component of the transition (if such exists).

The general algorithm (which we will denote by **Algorithm A"**) for tokens transfer after the time moment t_1 = TIME (the current IFGN2 time-moment) is as follows:

Algorithm A''

Step (A"01). The non-empty input places and the non-full output places are sorted by priority in a descending order. In comparison to **Algorithm A'**, each place may contain only one certain quantity of matter, that is analogous to a single token.

Step (A"02). *(Identical to step (A'03))* An empty index index matrix R, which corresponds to the index matrix of the predicates r of the given transition, is generated. It is initiated without values. After this, we put values "$\langle 0, 1\rangle$" (corresponding to value "false") of all of the elements of this index matrix, which:

– are placed in a row, corresponding to an empty input place, or
– are placed in a column, corresponding to a full output place, or
– are placed in (i, j)-th position, for which $m_{i,j} = 0$, i.e. the current capacity of the arc between i-th input and j-th output place is zero.

Step (A"03). The sorted places are passed sequentially by their priority, starting with the place having the highest priority, which contains some quantity of matter. In each of the passed places, the predicates corresponding to the relevant row of the index matrix R are checked. If the quantity of matter cannot split, the check finishes with finding the first predicate with truth-value different from "$\langle 0, 1\rangle$"; in the other case, we must calculate the truth values of all predicates in the row, for which the elements of R are not zeros.

Step (A"04). Depending on the execution of the operator for permission or prohibition of tokens splitting over the net, the token from **Step (A"03)** will pass either to all permitted to it output places, or to this very place among them, which has the highest priority, following one of the seven conditions below, where for predicate $r_{i,j}$ it holds that $f(r_{i,j}) = \langle \mu_{i,j}, \nu_{i,j}\rangle$:

C1 $\mu_{i,j} = 1,\ \nu_{i,j} = 0$ (the case for an ordinary GN);

C2 $\mu_{i,j} > \dfrac{1}{2}\ (> \nu_{i,j})$;

C3 $\mu_{i,j} \geq \dfrac{1}{2}\ (\geq \nu_{i,j})$;

C4 $\mu_{i,j} > \nu_{i,j}$;

C5 $\mu_{i,j} \geq \nu_{i,j}$;

C6 $\mu_{i,j} > 0$;

C7 $\nu_{i,j} < 1$, i.e., at least $\pi_{i,j} > 0$.

The condition, that will be used, is determined for each transition before the IFGN2-firing.

In comparison to the previous algorithm, the following additional specific requirement holds: $\sum_j \mu_{i,j} \leq 1$. The reason for this condition is that the quantity of matter in the i-th input place will be distributed to the j-th output place according to the truth degree $\mu_{i,j}$. In the i-th input place, there will remain such a quantity of matter that corresponds to the falsity degree $\nu_{i,j}$. Along the arc, connecting the i-th input and the j-th output place, there remains the rest quantity of matter corresponding to the supplement $\pi_{i,j} = 1 - \mu_{i,j} - \nu_{i,j}$.

Step (A"05). *(Identical to Step (A'06))* The values of the characteristic function for the output places (one or more), in which the tokens have entered (according to Step (A"04)), are calculated and the following record is assigned as a next token characteristic:

"value of function Φ for the respective token, $\mu_{i,j}, \nu_{i,j}$"

Step (A"06). *(Identical to Step (A'07))* Put values $\langle 0,1 \rangle$ in all rows of R for which the respective input place (from where the token goes out on Step (A"04)) is already empty; or in the columns of R which are full as a result of token's transfer on Step (A"04); or in the cells in the index matrix that correspond to arcs between the discussed input place and these output places, for which the arc-capacity becomes "0" as a result of the token's transfer.

Step (A"07). The current quantity of matter in each input place for the current transition decreases with the quantities of matter corresponding to $\sum_j \mu_{i,j}$. If the capacity of the place is empty, the elements of the corresponding row of index matrix R are assigned the value "0".

Step (A"08). The current quantity of matter in each output place for the current transition increases with new quantities of matter corresponding to $\sum_i \mu_{i,j}$. If the capacity of the place is reached, the elements of the corresponding column of the index matrix R are assigned value "0".

Step (A"09). The capacities of all arcs between the i-th input and the j-th output place, through which matter has transferred, decrement with quantity proportional to $\mu_{i,j} + \pi_{i,j}$. If the capacity of an arc has reached 0, the element from the index matrix R that corresponds to this arc is assigned the value $\langle 0, 1 \rangle$.

Step (A"10). Termination of the transition functioning.

The general algorithm for an IFGN's functioning coincides with the algorithm for an IFGN1's functioning.

The definition of FGNs of the second type (FGN2) can be given in an analogous way.

7. Intuitionistic Fuzzy Generalized Nets from third and fourth types (IFGN3s and IFGN4s)

Here, following [6, 7], we shall introduce IFGNs of type 3 (IFGN3s) and IFGN of type 4 (IFGN4s). They are extensions of IFGN1 and IFGN2s, respectively. Now, the tokens or places characteristics will be estimated in intuitionistic fuzzy sense, i.e., they will obtain values (intuitionistic fuzzy pairs), that will represent the degrees of validity and of non-validity of the characteristics. Therefore, the two new types of GNs allow describing situations where the model determines its status with degrees of validity and non-validity.

Every IFGN3s and each of its transitions has the forms from Section 3.

For instance, in IFGN1s, the values of function Φ are also estimated, i.e., now there are two real numbers $\mu(x_{\text{cu}}^{\alpha})$ and $\nu(x_{\text{cu}}^{\alpha})$, such that $\mu(x_{\text{cu}}^{\alpha}), \nu(x_{\text{cu}}^{\alpha}) \in [0, 1]$ and $\mu(x_{\text{cu}}^{\alpha}) + \nu(x_{\text{cu}}^{\alpha}) \le 1$. Only when these estimations satisfy the following conditions:

C1 $\mu(x_{\text{cu}}^{\alpha}) = 1$, $\nu(x_{\text{cu}}^{\alpha}) = 0$ (the case for an ordinary GN);

C2 $\mu(x_{\text{cu}}^{\alpha}) > \dfrac{1}{2}$ $(> \nu(x_{\text{cu}}^{\alpha}))$;

C3 $\mu(x_{\text{cu}}^{\alpha}) \ge \dfrac{1}{2}$ $(\ge \nu(x_{\text{cu}}^{\alpha}))$;

C4 $\mu(x_{\text{cu}}^{\alpha}) > \nu(x_{\text{cu}}^{\alpha})$;

C5 $\mu(x_{\text{cu}}^{\alpha}) \ge \nu(x_{\text{cu}}^{\alpha})$;

C6 $\mu(x_{\text{cu}}^{\alpha}) > 0$;

C7 $\nu(x_{\text{cu}}^{\alpha}) < 1$, i.e., at least $\pi(x_{\text{cu}}^{\alpha}) > 0$.

the tokens will obtain characteristics and now they will have the form

$$x_{\text{cu}}^{\alpha} = \left\langle \bar{x}_{\text{cu}}^{\alpha}, \mu(r_{i,j}), \nu(r_{i,j}), \mu(x_{\text{cu}}^{\alpha}), \nu(x_{\text{cu}}^{\alpha}) \right\rangle,$$

where $\bar{x}_{\text{cu}}^{\alpha}$ is the standard token's characteristic.

We proceed with the definition of the concept of IFGN4s. In this net, tokens are some "quantities" moving throughout the net.

The values of the transition condition's predicates can be intuitionistic fuzzy pairs, i.e., they can have degrees of truth and of falsity, as in the cases of IFGN1s and IFGN3s. The IFGN4 has the form

$$E = \langle\langle A, \pi_A, \pi_L, c, f, \theta_1, \theta_2\rangle, \langle K, \pi_K, \theta_K\rangle, \langle T, t^{\circ}, t^*\rangle, \langle X, \Phi, b\rangle\rangle$$

where A is the set of the net's transitions that have the ordinary GN-form with one difference: here the index matrix M contains as elements real numbers – capacities of the transition's arcs (as in the IFGN2's case).

The functions in the first component are similar to those of IFGN2 and they satisfy the same conditions.

The essential difference between IFGN2s and IFGN4s on one side, and the rest of GNs, on the other, is the set K of the GN-tokens and the functions related to it. Now the elements of K are some "quantities" that have only initial characteristics and that do not receive current characteristics. Function θ_K determines the time-moment when a given token will enter the net, as in the ordinary GNs.

The temporal components are also as in the other GN-types.

For both types of GNs IFGN2s and IFGN4s function Φ has a new meaning, similar to this of IFGS3. Now it assigns the places characteristics (the quantities of the tokens from the different times in the corresponding places) and their degrees of validity and of non-validity. As in GNs, it can be extended: it can give also other data about the modelled process (e.g., the time moments for the entry of the "quantities" in the places).

Similarly to IFGN2s and IFGN3s, function Φ in the IFGN4s is estimated too. However, here for every place p, this function gives the two real numbers $\mu(x_{\text{cu}}^{p})$

and $\nu(x_{\text{cu}}^p)$, such that $\mu(x_{\text{cu}}^p),\ \nu(x_{\text{cu}}^p) \in [0,1]$ and $\mu(x_{\text{cu}}^p)+\nu(x_{\text{cu}}^p) \le 1$. Now, the above conditions have the form:

C1 $\mu(x_{\text{cu}}^p) = 1,\ \nu(x_{\text{cu}}^p) = 0$ (the case for an ordinary GN);

C2 $\mu(x_{\text{cu}}^p) > \dfrac{1}{2}\ (> \nu(x_{\text{cu}}^p))$;

C3 $\mu(x_{\text{cu}}^p) \ge \dfrac{1}{2}\ (\ge \nu(x_{\text{cu}}^p))$;

C4 $\mu(x_{\text{cu}}^p) > \nu(x_{\text{cu}}^p)$;

C5 $\mu(x_{\text{cu}}^p) \ge \nu(x_{\text{cu}}^p)$;

C6 $\mu(x_{\text{cu}}^p) > 0$;

C7 $\nu(x_{\text{cu}}^p) < 1$, i.e., at least $\pi(x_{\text{cu}}^p) > 0$.

and the places will obtain characteristics and now they will have the form

$$x_{\text{cu}}^p = \left\langle \bar{x}_{\text{cu}}^p, \mu(r_{i,j}), \nu(r_{i,j}), \mu(x_{\text{cu}}^p), \nu(x_{\text{cu}}^p) \right\rangle,$$

where $\bar{x}_{\text{cu}}^p$ is the place's characteristic from IFGN2.

8. Conclusion

The algorithms above described are introduced here for the first time. They are optimized for faster performance of large IFGN models. In the near future the algorithms will be implemented in C^{++} and included in GN Ticker [10, 11], the main component of a current software package for GNs.

Acknowledgements: The authors are grateful for the support provided by projects DID-02-29 "Modeling Processes with Fixed Development Rules" and BIn-2/09 "Design and Development of Intuitionistic Fuzzy Logic Tools in Information Technologies" funded by the National Science Fund, Bulgarian Ministry of Education, Youth and Science.

References

1. Alexieva, J., E. Choy, E. Koycheva. Review and Bibliography on Generalized Nets Theory and Applications. – In: A Survey of Generalized Nets. E. Choy, M. Krawczak, A. Shannon and E. Szmidt, Eds. – In: Raffles KvB Monograph. Vol. **10**. 2007, 207-301.
2. Atanassov, K. Generalized Nets and Their Fuzzings. – AMSE Review, Vol. **2**, 1985, No 3, 39-49.
3. Atanassov, K. Generalized Index Matrices. – Compt. Rend. Acad. Bulg. Sci., Vol. **40**, 1987, No 11, 15-18.
4. Atanassov, K. Generalized Nets. Singapore, New Jersey, London, World Scientific, 1991.
5. Atanassov, K. Intuitionistic Fuzzy Sets. Heidelberg, Springer-Verlag, 1999.
6. Atanassov, K. On Generalized Nets Theory. Sofia, "Prof. M. Drinov" Academic Publishing House, 2007.
7. Atanassov, K., N. Nikolov. Intuitionistic Fuzzy Generalized Nets: Definitions, Properties, Applications. Systematic Organization of Information in Fuzzy Systems. P. Melo-Pinto, H.-N. Teodorescu and T. Fukuda, Eds. Amsterdam, IOS Press, 2003, 161-175.

8. R a d e v a, V., M. K r a w c z a k, E. C h o y. Review and Bibliography on Generalized Nets Theory and Applications. – Advanced Studies in Contemporary Mathematics, Vol. **4**, 2007, No 2, 173-199.
9. S t a r k e, P. Petri-Netze. Berlin, VEB Deutscher Verlag der Wissenschaften, 1980.
10. T r i f o n o v, T., K. G e o r g i e v. GN Ticker – A Software Tool for Efficient Interpretation of Generalized Net Models. – Issues in Intuitionistic Fuzzy Sets and Generalized Nets, Vol. **3**, Warsaw, 2005.
11. GN Ticker from Ifigenia, the Wiki on Intuitionistic Fuzzy Sets and Generalized Nets. **http://www.ifigenia.org/wiki/GNTicker**