Intelligent system modeling with total fuzzy grammars

RALF FABIAN, VASILE CRĂCIUNEAN, EMIL MARIN POPA
Department for Computer Sciences and Economic Informatics
"Lucian Blaga" University of Sibiu
Ion Rațiu Street, no. 5-7, Sibiu
ROMANIA

Abstract: - In this study we introduce the concept of total fuzzy grammars. Such grammars may be a useful tool for developing intelligent systems based on formal language theory. These reasoning can help us in generating words of the language with membership degrees to the maximal language.
In the end we present an algorithm for calculating membership degrees of terminal and nonterminal symbols. In this algorithm the interesting part is based on an extension operator design for auto-instructed stimulation. The result is a flexible tool witch is extendable to various tasks and domains. The goal of our approach here is to support the development of software products that have some knowledge and the ability of self instruction. An intelligent system must have a collection of information or a knowledgebase that enables him to treat information in such manner to obtain solutions for certain problems und to update these collections

Key-Words: - fuzzy grammar, total fuzzy grammar, fuzzy language, extension operator

1 Introduction

We assume familiarity with basic definitions and results of formal language theory. Therefore we present as basic definitions only some important aspects of fuzzy logics, needed in the formal modeling of intelligent systems presented in this paper.

A way of extending the concepts of formal languages to include soma aspects of natural processes is the introduction of fuzziness. Fuzzy logic is the logic behind approximate reasoning instead of exact reasoning. Its importance lies in the fact that many types of human reasoning, particularly the reasoning based on common sense, are by nature approximate. Lee and Zadeh [14] introduced fuzzy grammars as an extension of ordinary formal grammars [7] by using the concepts of fuzzy sets. There formal grammar is based on the Chomsky hierarchy.

Fuzzy grammars have only the terminal alphabet as fuzzy set. In this paper we introduce total fuzzy grammars and extend the membership function for the nonterminal alphabet, and finally establish a relationship between the syntactic analyze of a word from the language and his degree of membership to the language. Our approach here presents only the formalization and algorithm for the case of type 2 grammars, also known as context free grammars.

2 Basic definitions and notations

In this chapter we introduce some concepts that will support the construction of an intelligent software system.

Let X be a collection of objects called *universal set*. Let A be a regular (also called crisp) subset of X. For each x in X the characteristic function $f_A : X \to \{0,1\}$ determines whether x belongs or does not belong to A. Any such function, whose values are either zero or one, defines a crisp subset of X.

Fuzzy sets generalize the characteristic function in allowing all values between 0 and 1. A fuzzy subset F of X is defined by its membership function (a generalization of the characteristic function), also written $f_F : X \to [0,1]$, whose values can be any number in the interval [0, 1].

Definition 2.1. *A* ***fuzzy set*** *A over a universe of discourse* X *(a finite or infinite interval within which the fuzzy set can take a value) is a set of pairs*

$$A = \{\mu_A(x)/x \mid x \in X, \mu_A(x) \in [0,1] \in \mathbb{R}\}$$

where $\mu_A(x)$ *is called the* ***membership degree*** *of the element x to the fuzzy set A. This degree ranges between the extremes* **0** *and* **1** *of the dominion of the real numbers*:

- $\mu_A(x) = 0$ *indicates that x in no way belongs to the fuzzy set A.*
- $\mu_A(x) = 1$ *indicates that x completely belongs to the fuzzy set A.*

*Sometimes, instead of giving an exhaustive list of all the pairs that make up the set (**discreet values**), the membership degree is defined as function* $\mu_A : X \to [0,1]$, *referring to it as characteristic function or **membership function***. Crisp sets are considered special cases of fuzzy sets when membership values are always 0 or 1.

Grades of membership for universes other than the real numbers are normally calculated by the firing of rules, estimated by an observer or some other method, rather than by membership functions.

Members of a *numeric discrete fuzzy set* always describe a numeric quantity. Such discrete fuzzy sets are called *linguistic variables*, with members *linguistic terms*, for example "Speed" is a linguistic variable respectively to the linguistic terms "slow, medium, fast". A linguistic term is the word, in natural language, that expresses or identifies a fuzzy set that may or may not be formally defined. Thus, the membership function μ_A of a fuzzy set *A* expresses the degree in which *x* verifies the category specified by *A*.

The theory of fuzzy sets generalizes the theory of classic sets, this means that the fuzzy sets allow operations of union, intersection, and complement. These and other operations can be found in [8] and [10].

Definition 2.2. *Let A and B be two fuzzy sets over* X. *Then A is **equal** to B if*
$\forall x \in X$, $\mu_A(x) = \mu_B(x)$, denoted by $A = B$.

Definition 2.3. *Taking two fuzzy sets A and B over* X, *A is said to be **included** in B if*
$\forall x \in X$, $\mu_A(x) \le \mu_B(x)$, denoted by $A \subseteq B$.

Definition 2.4. A *Triangular Norm* (***t-norm***) *is a binary operation,* $\mathrm{T} : [0,1] \times [0,1] \to [0,1]$ *that complies with the following properties*:

a) Commutativity: $\mathrm{T}(x, y) = \mathrm{T}(y, x)$.
b) Associativity: $\mathrm{T}(x, \mathrm{T}(y, z)) = \mathrm{T}(\mathrm{T}(x, y), z)$.
c) Monotonicity: If $x \le y$, and $w \le z$ then $x\mathrm{T}w \le y\mathrm{T}z$.
d) Boundary conditions: $\mathrm{T}(x,0) = 0$ and $\mathrm{T}(x,1) = x$.

Definition 2.5. A *Triangular Conorm* (***t-conorm***) *is a binary operation,* $\perp : [0,1] \times [0,1] \to [0,1]$ *that complies with the following properties*:

a) Commutativity: $\perp(x, y) = \perp(y, x)$.
b) Associativity: $\perp(x, \perp(y, z)) = \perp(\perp(x, y), z)$.
c) Monotonicity: If $x \le y$, and $w \le z$ then $x \perp w \le y \perp z$.
d) Boundary conditions: $\perp(x,0) = x$ and $\perp(x,1) = 1$.

Definition 2.6. *If A and B are two fuzzy sets over a universe of discourse* X, *the membership function of the **union** of the two sets* $A \cup B$ *is expressed by*

$$\mu_{A \cup B}(x) = f(\mu_A(x), \mu_B(x)),\ x \in X \qquad (1)$$

where *f* is a t-conorm [15].

Definition 2.7. *If A and B are two fuzzy sets over a universe of discourse* X, *the membership function of the **intersection** of the two sets* $A \cap B$, *is expressed by*

$$\mu_{A \cap B}(x) = g(\mu_A(x), \mu_B(x)),\ x \in X \qquad (2)$$

where *g* is a t-norm [15].

The most widely used of this type of functions are the **t-norm of the Minimum** and the **t-conorm of the Maximum** as they have retained a large number of the properties of the Boolean operators. Thus, we consider in this paper the following form of the relations (1) and (2):

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)),\ x \in X \qquad (3)$$

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)),\ x \in X \qquad (4)$$

Between t-norm and t-conorm exists a relationship in form o an extension of De Morgan's Law:

$$\perp(x, y) = 1 - \mathrm{T}(1 - x, 1 - y)$$

$$\mathrm{T}(x, y) = 1 - \perp(1 - x, 1 - y)$$

Definition 2.8. For a fuzzy set *A* in the universe of discourse *X,* the membership function of the complement, denoted by $\overline{A}$ is shown as

$$\mu_{\overline{A}}(x) = 1 - \mu_A(x),\ x \in X. \qquad (5)$$

Contrary to the classical set theory, the following relations hold: $\overline{A} \cap A \ne \varnothing$ and $\overline{A} \cup A \ne X$.

3 Fuzzy languages and total fuzzy grammars

A *numerical variable* takes numerical values: "Age = 70". A linguistic variables takes linguistic values: "Age is old".

A *linguistic values* is a fuzzy set and has a *label* and a *meaning*.

label: Symbol, Sentence in a Language

meaning: Fuzzy Subset of a Universe of Discourse

A language can be seen as a fuzzy relation from a set of labels *T* to a Universe of Discourse *U*, which assigns a grade of membership $\mu_L(t,u)$ to each pair $(t,u) \in T \times U$. If we fix the term "t" say t_i, then $\mu_L(t_i,u)$ determines the fuzzy subset $A(t_i)$ whose membership function is $\mu_{A(t_i)}(u) = \mu_L(t_i,u)$. $\mu_{A(t_i)}(u)$ is the *meaning* of the linguistic value whose *label* is t_i. If we fix the term *u*, say u_j, then

$\mu_L(t,u_j)$ is the *descriptor* of u_j.

A language is a fuzzy mapping $U \times T \to [0,1]$.

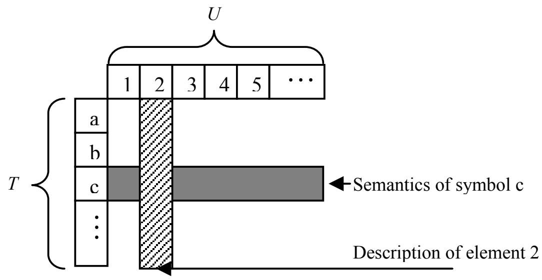

Fig. 1. Representation of fuzzy languages.

Fuzzy grammars [14] ore linguistic rules are used to describe the syntax of languages or the structural relation of patterns. They can also be used to characterize a syntactic source which generates all the sentences in a language [7].

Definition 3.1. *Let V be a countable, nonempty fuzzy subset of the universe of discourse X, named* ***fuzzy alphabet****. A fuzzy symbol is an element of the set V and a word over the alphabet V is a sequence of fuzzy symbols* $p = a_1a_2...a_n$, $a_i \in V$, $i = \overline{1,n}$.

The fuzzy alphabet or vocabulary V is defined through a membership function who maps to any $x \in X$ the degree of membership $\mu_V(x)$ between $[0,1]$, $\mu_V : X \to [0,1]$.

In the particular case when μ_V takes only values of 0 an 1, the fuzzy vocabulary *V* represents a classical subset of *X*.

We introduce now an extension operator defined as follows:

Definition 3.2. *An* ***extension operator*** *is a function* $\mathrm{E}:[0,1]\times[0,1]\to[0,1]$ *that satisfy for* $\forall x,y,z \in [0,1]$ *the following properties:*

a) Commutativity:
$$\mathrm{E}(x,y) = \mathrm{E}(y,x). \tag{6}$$
b) Associativity:
$$\mathrm{E}(x,\mathrm{E}(y,z)) = \mathrm{E}(\mathrm{E}(x,y),z). \tag{7}$$
c) Monotonicity:
If $x \le y$, and $w \le z$ then $x\mathrm{E}w \le y\mathrm{E}z$. (8)
d) Boundary conditions:
$$\mathrm{E}(x,0) = 0 \text{ and } \mathrm{E}(x,1) = x. \tag{9}$$

If c) and d) are satisfied then E is a t-norm, or if the following properties hold

e) Monotonicity:
If $x \le y$, and $w \le z$ then $x \perp w \le y \perp z$. (10)
f) Boundary conditions:
$$\perp(x,0) = x \text{ and } \perp(x,1) = 1. \tag{11}$$

then E is a t-conorm.

Consequently t-norms and t-conorms are extension operators.

Definition 3.3. Let *V be a fuzzy alphabet over de universe of discourse X, defined through the membership function* $\mu_V : X \to [0,1]$, *and the extension operator* $\mathrm{E}:[0,1]\times[0,1]\to[0,1]$. *Then the membership function* $\mu_L : X^* \to [0,1]$, *for any* $L \subseteq X^*$, *is defined as follows*:

- $\mu_L(a) = \mu_V(a)$, if $a \in V$; (12)
- $\mu_L(pa) = \mathrm{E}(\mu_L(p),\mu_V(a))$, if $p \in X^*$, $a \in V$. (13)

Definition 3.4. *A* ***fuzzy language*** *over the fuzzy alphabet V is a fuzzy subset* $L \subseteq V^*$ *defined through the membership function* $\mu_L : X^* \to [0,1]$. *In these case*

$$\mu_L(pq) = \mathrm{E}(\mu_L(p),\mu_L(q)),\ p,q \in X^*. \tag{14}$$

Definition 3.5. *Let* L_1 *and* L_2 *be to fuzzy grammars. The concatenation operation of* L_1 *and* L_2 *produce a fuzzy language* $L = L_1L_2$ *whit the membership function defined as*

$\mu_L(pq) = \mathrm{E}(\mu_{L_1}(p),\mu_{L_2}(q))$ if $p \in L_1$ and $q \in L_2$.

Hence, fuzzy languages are fussy sets all operations on fuzzy sets still hold.

Definition 3.6. *A* ***total fuzzy generative grammar*** *(or shortly total fuzzy grammar) is a sextuple* $FG = (V_N, V_T, S, P, \mu_{V_T}, \mathrm{E})$ *where*:

- V_N is the set of nonterminal symbols;
- V_T is the alphabet of terminal symbols, $V_N \cap V_T = \varnothing$;
- $S \in V_N$ is the starting symbol;
- P is a finite set of production rules of the form $\alpha \to \beta$, where $\alpha,\beta \in (V_N \cup V_T)^*$. α must contain at least one symbol from V_N;
- $\mu_{V_T} : X \to [0,1]$ is the membership function that specifies the fuzzy alphabet V_T;
- $\mathrm{E}:[0,1]\times[0,1]\to[0,1]$ is the extension operator.

Definition 3.7. The ***language generated by the fuzzy grammar*** $FG = (V_N, V_T, S, P, \mu_{V_T}, \mathrm{E})$ is the fuzzy subset

$$L(FG) = \{p \mid p \in V_T^* \text{ and } S \overset{*}{\Rightarrow} p, \mu_{V_T^*}(p) > 0\}$$

where $\mu_{V_T^*} : X^* \to [0,1]$ is the extension of μ_{V_T} (this extension for strings of elements is analogues to definition 3.4).

4 Membership function extension for type 2 grammars

As we have seen in the above, fuzzy grammars have only the terminal alphabet as fuzzy set. The problem that rises from here is to extend the membership function for the nonterminal alphabet to, and establish a relationship between the syntactic analyze of a word from the language and his degree of membership to the language.

These reasoning can help us in generating words of the language with membership degrees to the maximal language. Intuitively speaking, we know that the membership degree of a nonterminal is not a constant and depends on the derivation in which he is involved. Thus, for each nonterminal we will have a finite or infinite set of possible membership degrees in extend to the derivation implied in.

We denote $V = V_T \cup V_N$ and extend the membership function $\mu_{V_T} : X \to [0,1]$ to the function $\mu_V : X \to \mathcal{P}([0,1])$, where $\mathcal{P}([0,1])$ is the set of partial subsets of the real interval $[0,1]$, as follows:

a) If $x \to r \in P$, where $r \in V_T^*$, than we have $r = v_0 v_1 ... v_n$, where $v_0, v_1, ..., v_n \in V_T$, $n \in \mathbb{N}$ and can define $\mu_V(x) = \mu_{V_T^*}(v_0 v_1 ... v_n)$.

b) If $x \to r \in P$, where $r \in (V_N \cup V_T)^*$, then $r = v_0 x_1 v_1 ... x_n v_n$, where $v_0, v_1, ..., v_n \in V_T^*$ and $x_1, x_2 ..., x_n \in V_N$. Thus, $\mu_V(x)$ can be defined only if $\mu_V(x_1), \mu_V(x_2), ..., \mu_V(x_n)$ are already defined.

Finally, the membership function $\mu_{V_T}(x)$ extends recursively to the function $\mu_V(x)$ as follows:

a) If $x \in V_T$ then $\mu_V(x) = \mu_{V_T}(x)$.

b) If $x \in V_N$ then $\mu_V = \mu_V(x_1), \mu_V(x_2), ..., \mu_V(x_n)$ for al productions of from $x \to x_1, x_2, ..., x_n$.

In any type 2 grammar with no useless symbols, for a nonterminal *x* exists at least one derivation of the form $x \overset{*}{\Rightarrow} r$ where $r \in V_T^*$. On the other hand, if there exists a derivation of the form $x \overset{*}{\Rightarrow} r$, $r \in V_T^*$ then the membership degree $\mu_V(x)$ can be calculated for the nonterminal *x*.

We now present an algorithm for calculating the membership degrees of al nonterminal symbols of a fuzzy grammar.

Algorithm. NMDCA (Nonterminal membership degree calculus algorithm)

Input: a fuzzy grammar of type 2.
Output: the membership function μ_V.
Method: Consists of constructing a series of sets $\{K_n \mid n \in \mathbb{N}\}$, having the property $K_0 \subset K_1 \subset ... \subset K_{h-1} \subset K_h$. As it can be easily seen, K_h will contain all nonterminal symbols whose membership degrees have already been calculated.

01 **START**
02 $j := 0$
03 $K_0 := \varnothing$
04 **IF** $x \in V_T$ **THEN** $\mu_V(x) = \mu_{V_T}(x)$
05 **DO UNTIL** ($K_j = K_{j-1}$)
06 $\mu_V(x) = \mu_V(x) \cup$ $\cup \{\mu_{V_T}(r) \mid x \to r \in P \, and \, r \in (K_j \cup V_T)^*\}$
07 $K_{j+1} := K_j \cup$ $\cup \{x \mid x \to \Gamma \in P \, and \, \Gamma \in (K_j \cup V_T)^*\}$
08 $j := j + 1$
09 **ENDDO**
10 **STOP**

When the algorithm sops the set K_j will be equal to V_N, only if the grammar has no useless symbols. Thus, the algorithm calculates the membership degrees for all nonterminal symbols.

We notice that a nonterminal may have more then one membership degree, this means each α-production ($\alpha \to \beta$, $\alpha \in V_N$, $\beta \in (V_N \cup V_T)^*$) has another degree witch is considered as membership degree of the production. In other words, every α-production has his membership degree.

Theorem. If *G* is a fuzzy grammar of type 2 then the membership function $\mu_{V_T} : X \to [0,1]$ can be extended to the membership function $\mu_V : X \to \mathcal{P}([0,1])$, where $\mathcal{P}([0,1])$ is the set of partial subsets of the real interval $[0,1]$.

Proof. From the mode of defining the sets K_n in line 07 of the NMDCA, we have $K_n \subseteq V_N$, $(\forall) n \in \mathbb{N}$ and $\{K_n \mid n \in \mathbb{N}\}$ is an increasing sequence of sets. Hence V_N is a finite set, we can build at most $|V_N| + 1$ sets, respectively for $h \le |V_N|$ we have $K_0 \subset K_1 \subset ... \subset K_{h-1} \subset K_h = K_{h+1} = ...$.

Example 4.1 Lets consider a grammar for simple house drawing [20] with the following elements: V_N={<house>, <facade>, <front>, <sideview>,

<wall>, <window>, <door>,<chimney>, <roof>};
V_T ={in, inW, inN, inS, inE, inNW, inSW, inNE, inSE, outN, outS, outE, outNE, outSE, intersect, “,” , square, triangle, rhomboid, trapezoid}.

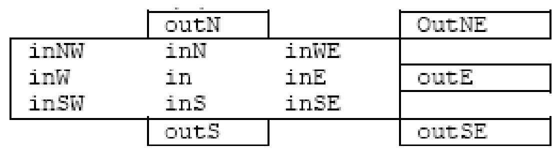

Fig. 2. Positioning terminal Symbols.

I, W, N, S, denote the four cardinal points and "in" and "out" denote that an object is
inside or outside another object. Other position terminal symbols are "intersect" for intersection, "(...)" for enclosure, etc.

S = <house> and
P:
<house> (1.0)→ <sideview> outE <house> ;
<house> (1.0)→ <facade> ;
<door> (1.0)→ square ;
<window> (0.8)→ square ;
<window> (1.0)→ square outE square ;
<chimney> (0.8)→ trapezoid ;
<chimney> (0.9)→ square ;
<wall> (1.0)→ square ;
<wall> (1.0)→ (<window> in square) ;
<wall> (0.8)→ (<window> <door> in square) ;
<wall>(0.95)→ (<door> inS square) ;
<front> (1.0)→ triangle ;
<front> (1.0)→ <chimney> outS triangle ;
<roof> (1.0)→ rhomboid ;
<roof> (1.0)→ <chimney> outS rhomboid ;
<facade> (0.95)→ <front> outS <wall> ;
<sideview> (1.0)→ (<roof> outS <wall>) .

The proposition (square outS rhomboid outS (square in square)) outE (triangle outS (square inS square)) may generate the following drawing:

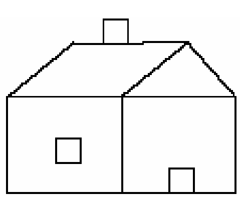

Fig. 3. Graphical representation of the proposition from example 4.1.

meaning the house is recognized with grade of membership 0.8.

5 Conclusions and future research directions

The great potential that the use of membership degrees represents is to allow something qualitative (fuzzy) to be expressed quantitatively by means of the membership degree. The greatest benefit of using fuzzy grammars we may have in cases with syntactical ambiguity, meaning an input sequence can be analyzed by two distinct syntactic trees. Here the degrees of membership respectively the method of evaluating them can take a clearly decision. On the other had, in fuzzy grammar theory we may have grammatical incorrect propositions, due the fact that the membership degree is bigger then 0.

The goal of our approach here is to support the development of software products that have some knowledge and the ability of self instruction.

In case of a software system for production process simulation the grammar vocabulary would be the set of elementary operations of the process, that run on an activity subject level. In each moment follows the execution of an elementary operation caused by a decision. If these operations have a specific degree of membership to the grammar vocabulary, the decision can be taken based on this membership degree. Moreover, according to the outcome of a decision, other decisions can be stimulated by changing there membership degree of the corresponding operations. Thus the system may learn upon his experience.

References:
[1] Ralf Fabian, *Limbaje formale, Teorie, Exemple, Probleme*, Editura Univeristății „Lucian Blaga” Sibiu, 2006.
[2] Crăciunean V., *Translatoare şi compilatoare, Ed. Alma Mater*, Sibiu, 2002.
[3] Emil M. Popa, *Modele formale computationale, Editura* „Alma Mater”, Sibiu, 2000
[4] Emil M. Popa, *Limbaje formale. Fundamentele limbajelor de programare*, Editura „Alma Mater”, Sibiu, 2003
[5] Emil M. Popa, *Programare genetica si evolutiva*, Editura „Alma Mater”, Sibiu, 2003
[6] Emil M. Popa, *Formal Syntax and Semantics of Programming Language*, Editura „Alma Mater”, Sibiu, 2004
[7] Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, *Compilers: Principles, Techniques, and Tools*, Addison Wesley, 2001.
[8] K. S. Fu., *Syntactic Pattern Recognition and applications*, NJ, Prentice-Hall, 1982.

[9] Petry, F. E. (with Bose, P.). (1996). *Fuzzy databases: Principles and applications*. In H. J. Zimmermann (Ed.), International series in intelligent technologies. Kluwer Academic.

[10] Pedrycz, W., & Gomide, F. (1998). *An introduction to fuzzy sets: Analysis and design.* Cambridge, MA: MIT Press.

[11] Zadeh, L. A. (1965). *Fuzzy sets. Information and Control*, 8, 338-353.

[12] Zadeh, L. A. (1971). *Similarity relations and fuzzy orderings*. Information Sciences, 3, 177-200.

[13] Zadeh, L. A. (1992). *Knowledge representation in fuzzy logic. In An introduction to fuzzy logic applications in intelligent systems*. Kluwer Academic.

[14] Lee E., Zadeh L. – *Note on Fuzzy Languages*, Information Sciences, vol. 1, p. 421-434, 1969

[15] Schweizer, B., & Sklar, A. (1983). *Probabilistic metric spaces*. North-Holland, Amsterdam.

[16] William Siler, James J. Buckley - *Fuzzy Expert Systems and Fuzzy Reasoning*, Published by John Wiley & Sons, Inc., Canada, 2005.

[17] Jose Galindo, Angelica Urrutia, Mario Piattini - Fuzzy *Databases: Modeling, Design and Implementation*, Idea Group Publishing, USA, 2006.

[18] Christian W. Omlin, Karvel K. Thornber, C. Lee Giles - *Representation of Fuzzy Finite State Automata in Continuous Recurrent Neural Networks*, Proceedings of IEEE International Conference on Neural Networks, p. 1023, IEEE Press, 1996.

[19] Klir, G. J., and Yuan, B. *Fuzzy Sets and Fuzzy Logic : Theory and Applications*. Prentice Hall (Sd), 1995.

[20] Rabitti F., Stanchev P., *An Approach to Image Retrieval from Large Image Databases*, Proc. ACM-SIGIR Conf., New Orleans, pp.284-295, 1987.

[21] Peter L. Stanchev, David Green Jr., *Formal Languages for Image Primitives Description*, Kettering University, Flint.

[22] Teodor Rus, *Mecanisme formale pentru specificarea limbajelor*, Ed. Academie Române, Bucureşti, 1983.

[23] Creangă, C. Reischer, D. Simovici, *Introducere algebrică în informatică – Limbaje formale*, Ed. Junimea, Iaşi, 1974.

[24] Creangă, C. Reischer, D. Simovici, *Introducere algebrică în informatică – Teorie automatelor*, Ed. Junimea, Iaşi, 1973.

[25] Toader Jucan, *Limbaje formale şi automate*, Ed. MatrixRom, 1999.

[26] Gabreil V. Orman, *Limbaje formale şi acceptori*, Ed. Albastră, Cluj-Napoca, 2002.

[27] Luca Dan Şerbănaţi, *Limbaje de programare şi compilatoare*, Ed. Academiei Române, Bucureşti, 1987.

[28] Gh. Păun, *Mecanisme generative ale proceselor economice*, Ed Tehnică, Bucureşti, 1988.