MTA SZTAKI Közlemények 33/1985 p. 55-67

ON FUZZY AUTOMATA AND FUZZY GRAMMARS

K.G.Peeva

Center of Applied Mathematics
Sofia 1000,P.O.Box 384

ABSTRACT. Fuzzy grammars and fuzzy languages in connection with finite fuzzy acceptors are studied. Let A be a finite fuzzy acceptor and R(A) be the set of all words recognizable by A. It is proved that for each fuzzy regular grammar G_F generating the language $L(G_F)$ there exists a finite fuzzy acceptor A such that $R(A)=L(G_F)$ and vice versa.

The main results are about algorithmical decidability of ε-equivalence and ε-reduction by inputs. It is shown that the relation ε-closeness of matrices is invariant. On this base some properties of the ε-equivalence and ε-reduction are obtained and their application in syntactic pattern recognition are discussed.

1. ε-CLOSENESS OF MATRICES

In this section ε-closeness for matrices over a bounded chain is defined and studied. These algebraic results are necessary for the ε-equivalence and ε-reduction by input words which is the subject of section 3. The algebraic terminology is according to [3].

Let $\mathbb{L}=([0,1],\vee,\wedge,0,1)$ be a bounded chain [3] over the ordered set $[0,1]\subset\mathbb{R}$ with lower and upper bounds respectively 0 and 1 and operations $\vee$ and $\wedge$.

Let $A=(a_{ij})_{mxn}$ and $B=(b_{ij})_{nxp}$ be matrices over the bounded chain $\mathbb{L}$ with elements $a_{ij},b_{ij}\in[0,1]$ for each i,j. The matrix $C=AB=(c_{ij})_{mxp}$ is a *product* of A and B if

$$c_{ij} = \overset{n}{\underset{k=1}{\vee}} (a_{ik} \wedge b_{kj})$$ for each $i=1,\dots,m$ and each $j=1,\dots,p$ [4].

It is easily established that the matrix multiplication is associative. Having in mind this property we shall omit the breckets next.

Let $A=(a_{ij})$ and $B=(b_{ij})$ be mxn-matrices and $\varepsilon\in[0,1]$ be fixed. We say that the i^{th} row in A is *ε-close* to the k^{th} row in B if $|a_{is}-b_{ks}|\leq\varepsilon$ holds for each s, $1\leq s\leq n$; A and B are *ε-close* (notation $d(A,B)\leq\varepsilon$) if $|a_{ij}-b_{ij}|\leq\varepsilon$ holds for each i, $1\leq i\leq m$ and for each j, $1\leq j\leq n$.

Theorem 1. If $A=(a_j)_{nx1}$ and $B=(b_j)_{nx1}$ are ε-close then $|\overset{\vee}{_j}(a_j)-\overset{\vee}{_j}(b_j)|\leq\varepsilon$ is valid.

Proof. Let $\overset{\vee}{_j}(a_j)=a_k$ and $\overset{\vee}{_j}(b_j)=b_r$. Then

$$a_k-\varepsilon\leq b_k\leq b_r\leq a_r+\varepsilon\leq a_k+\varepsilon \implies$$
$$a_k-\varepsilon\leq b_r\leq a_k+\varepsilon \implies$$
$$-\varepsilon\leq b_r-a_k\leq\varepsilon \iff |b_r-a_k|\leq\varepsilon, \text{ i.e. } |\overset{\vee}{_j}(a_j)-\overset{\vee}{_j}(b_j)|\leq\varepsilon$$

In particular, if $|a_k-a_i|\geq2\varepsilon$ for each $i\neq k$ then $\overset{\vee}{_j}(a_j)=a_k$ and $\overset{\vee}{_j}(b_j)=b_k$ for the same index k because

$$a_i+2\varepsilon\leq a_k \iff a_i+\varepsilon\leq a_k-\varepsilon \implies b_i\leq a_i+\varepsilon\leq a_k-\varepsilon\leq b_k$$ and hence $b_i\leq b_k$ for each $i\neq k$.

It is easy to see that Th.1 is valid for A^t and B^t as well.

Theorem 2. If $d(A,B)\leq\varepsilon$ and $d(C,D)\leq\varepsilon$ then $d(AC,BD)\leq\varepsilon$ holds whenever the products make sence.

Proof. According to the definitions $d(AC,BD)\leq\varepsilon \iff$ $|\overset{\vee}{_k}(\wedge(a_{ik},c_{kj}))-\overset{\vee}{_k}(\wedge(b_{ik},d_{kj}))|\leq\varepsilon$ for each i,j. We shall prove the last inequation for arbitrary i,j. For the vector-matrices

$$V_{ij} = (v_{ij}(k)) = (\wedge(a_{i1},c_{1j}),\wedge(a_{i2},c_{2j}),\dots)$$
$$W_{ij} = (w_{ij}(k)) = (\wedge(b_{i1},d_{1j})\ \wedge(b_{i2},d_{2j}),\dots)$$

we obtain $d(V_{ij},W_{ij})\leq\varepsilon$ because $|v_{ij}(k)-w_{ij}(k)|\leq\varepsilon$ for each k as it is shown by the following points:

1. If $v_{ij}(k)=\hat{}(a_{ik},c_{kj})=a_{ik}$ and $w_{ij}(k)=\hat{}(b_{ik},d_{kj})=b_{ik}$ then $|v_{ij}(k)-w_{ij}(k)|=|a_{ik}-b_{ik}|\leqq\varepsilon$ because $d(A,B)\leqq\varepsilon$;

2. If $v_{ij}(k)=\hat{}(a_{ik},c_{kj})=c_{kj}$ and $w_{ij}(k)=\hat{}(b_{ik},d_{kj})=d_{kj}$ then $|v_{ij}(k)-w_{ij}(k)|=|c_{kj}-d_{kj}|\leqq\varepsilon$ because $d(C,D)\leqq\varepsilon$;

3. If $v_{ij}(k)=\hat{}(a_{ik},c_{kj})=a_{ik}$ and $w_{ij}(k)=\hat{}(b_{ik},d_{kj})=d_{kj}$ then $b_{ik}\in[a_{ik}-\varepsilon,a_{ik}+\varepsilon]\subset[0,1]$; $c_{kj}\in[a_{ik},1]$ because $a_{ik}\leqq c_{kj}$. Since $d_{kj}\in[c_{kj}-\varepsilon,c_{kj}+\varepsilon]$ and $d_{kj}\in[0,a_{ik}+\varepsilon]$ $(d_{kj}\leqq b_{ik}\leqq a_{ik}+\varepsilon)$ we obtain $d_{kj}\in[a_{ik}-\varepsilon,a_{ik}+\varepsilon]$ and hence $|v_{ij}(k)-w_{ij}(k)|=|a_{ik}-d_{kj}|\leqq\varepsilon$

4. If $v_{ij}(k)=\hat{}(a_{ik},c_{kj})=c_{kj}$ and $w_{ij}(k)=\hat{}(b_{ik},d_{kj})=b_{kj}$ by analogy with the previous case we obtain $|v_{ij}(k)-w_{ij}(k)|\leqq\varepsilon$

Since $|v_{ij}(k)-w_{ij}(k)|\leqq\varepsilon$ is valid for each k we have $d(V_{ij},W_{ij})\leqq\varepsilon$. According to Th.1 the inequation $|\check{}_k\, v_{ij}(k)-\check{}_k\, w_{ij}(k)|\leqq\varepsilon$ is true, i.e. $|\check{}_k(\hat{}(a_{ik},c_{kj}))-\check{}_k(\hat{}(b_{ik},d_{kj}))|\leqq\varepsilon$ is valid.

Theorem 3. If $d(A,B)\leqq\varepsilon$ then:

i) $d(CA,CB)\leqq\varepsilon$; ii) $d(AT,BT)\leqq\varepsilon$; iii) $d(CAT,CBT)\leqq\varepsilon$ whenever the products make sence.

The proof follows from Th.2.

2. FUZZY ACCEPTORS AND FUZZY GRAMMARS

We define and study fuzzy acceptors and fuzzy grammars by analogy with [2] where the stochastic acceptors and stochastic grammars are considered. The terminology for automata and language theories is according to [2],[4],[5].

A fuzzy automaton A [4],[6] is a quintuple $A=(X,Q,Y,M,\mathbb{L})$ where:

(i) X,Q,Y are nonempty sets of input letters, states and output letters respectively;

(ii) $M=\{M(x/y)=(m_{ij}(x/y))/x\in X, y\in Y, m_{ij}\in[0,1]\}$ is the set of the transition-output matrices, the step-wise behaviour of A;

(iii) $\mathbb{L}=([0,1]^{\vee},{}^{\wedge}0,1)$ is the bounded chain.

If X,Q,Y are finite then A is called *finite automaton.*

The interpretation of the membership degrees $m_{ij}(x/y)\in[0,1]$ is well-known [4],[6]: each element $m_{ij}(x/y)$ determines the step-wise behaviour of A. If in step t the automaton is in state q_i and receives the input letter x, it puts out the output letter y in step t and reaches the state q_j in the next step t+1 with the membership degree $m_{ij}(x/y)\in[0,1]$.

A finite fuzzy acceptor (shortly *acceptor*) $A=(X,Q,q_o,F,M,\mathbb{L})$ is a finite fuzzy automaton without outputs (i.e. $|Y|=1$), with fixed initial state $q_o\in Q$ and with a set $F\subset Q$ of the final states.

Let X* be the free monoid generated by X with $e\in X^*$ as unit element. We extend the step-wise behaviour of the acceptor $A=(X,Q,q_o,F,M,\mathbb{L})$ to the complete behaviour of A for $k\in\mathbb{N}$ consecutive steps as follows: since the empty word e need no time we define M(e)=I, where I stands for the unitary matrix of order $|Q|$, the cardinality of Q; if the input word $u\in X^*$ is a letter $x\in X$ then the transition matrix is M(u)=M(x); if in k>1 consecutive steps the letters $x_1,\dots,x_k\in X$ are fed into A (i.e. the input word is $u=x_1\dots x_k\in X^*$) then $M(u)=M(x_1)\dots M(x_k)$ and an arbitrary element $m_{ij}(u)$ in M(u) is interpreted as the membership degree for the state q_i and the input word u in step t under teh state q_j in step t+k. We denote by M* the set of all transition matrices (the complete behaviour) for the given acceptor A:

$$M^* = \{M(u) = (m_{ij}(u))/u\in X^*\}.$$

The set of all words $u\in X^*$, which are recognizable by the acceptor A is denoted by R(A):

$R(A) = \{u/u \in X^*, \exists m_{oj}(u)>0, q_j \in F\}$.

We define the notion fuzzy grammar by analogy with [2] where stochastic grammars are defined and studied.

A fuzzy grammar $G_F=(N,T,S,P_F)$ is specified by a finite set N of nonterminal symbols, a finite set T of terminal symbols, disjoint from N, an element $S \in N$ called the start symbol and a finite set of fuzzy productions

$a_i \xrightarrow{p_{ij}} b_{ij}$, $i=1,\dots,k$, $j=1,\dots,n_i$, where $a_i \in (N \cup T)^* N (N \cup T)^*$, $b_{ij} \in (N \cup T)^*$ and $p_{ij} \in [0,1]$ is the mebership degree for this production. For the fuzzy grammar G_F we say that w *directly derives* w' (notation $w \xrightarrow{p_{ij}} w'$) with the membership degree p_{ij} iff $w=c_1 a_i c_2$, $w'=c_1 b_{ij} c_2$ and $a_i \xrightarrow{p_{ij}} b_{ij}$ is a production in P_F; we say that w *derives* w' with membership degree $p=\bigwedge_j p_j$ (notation $w \xrightarrow[*]{p} w'$) if there exists a sequence $w_1,\dots,w_{n+1}$ in $(N \cup T)^*$ such that $w=w_1$, $w'=w_{n+1}$ and $w_j \xrightarrow{p_j} w_{j+1}$ for $1 \leq j \leq n$. The binary relation $\xrightarrow[*]{}$ is the reflexive and transitive closure of $\rightarrow$.

The *fuzzy language* $L(G_F)$ generated by G_F is the set of all terminal strings which can be derived from S:

$$L(G_F) = \{(u, p(u))/u \in T^*, S \xrightarrow[*]{p_j} u, j=1,\dots,k, p(u)=\bigvee_{j=1}^{k} p_j>0\}.$$

The number of the different ways to obtain u from S is denoted by k.

Example 1. Let the fuzzy grammar $G_F=(N,T,S,P_F)$ with

$N = \{S_o,S_1,S_2\}$, $T=\{a,b\}$, $S=S_o$ and P_F:

$S_o \xrightarrow{0,1} aS_1$ $\qquad S_1 \xrightarrow{0,2} aS_1$ $\qquad S_1 \xrightarrow{0,7} a$

$S_o \xrightarrow{0,3} bS_2$ $\qquad S_1 \xrightarrow{0,5} bS_2$ $\qquad S_2 \xrightarrow{0,6} b$

be given. The fuzzy language $L(G_F)$ generated by G_F is

$$L(G_F)=\{(b^2,0,3)\} \cup \{(a^nb^2,0,1)/n\geq 1\} \cup \{(a^n,0,1)/n>1\}.$$

The grammar $G=(N,T,S,P)$ obtained from the fuzzy grammar $G_F=(N,T,S,P_F)$ by forgetting the membership degrees in the productions in P_F is called *associated* to G_F. The fuzzy grammar G_F has *type* 0,1,2,3 if its associated grammar G has type 0,1,2,3 respectively.

In this paper we consider only the fuzzy grammars of type 3. For a fuzzy grammar of type 3 (finite state, regular) all productions in P_F are as follows: $S_i \xrightarrow{p_{ij}} xS_j$ or $S_i \xrightarrow{p_i} y$, where $S_i,S_j \in N, x,y \in T$.

Theorem 4. Let $G_F=(N,T,S,P_F)$ be a fuzzy grammar of type 3. If $A=(X,Q,q_o,F,M,\mathbb{L})$ is an acceptor with $X=T$; $Q=N\cup\{E\}$; $q_o=S$; $F=\{S,E\}$ if the production $S \xrightarrow{p} e$ belongs to P_F and $F=\{E\}$ otherwise and the following membership degrees for each $q_i,q_j \in N$, $q_f \in F$, $x \in T$:

$m_{ij}(x)=p_{ij}>0$ if $q_i \xrightarrow{q_{ij}} xq_j$ is a production in P_F;

$m_{ij}(x)=p_i>0$ if $q_i \xrightarrow{p_i} x$ is a production in P_F;

$m_{ij}(x)=0$ otherwise,

then $R(A)=L(G_F)$.

Proof. Let $z_o=(z_o(i))_{1x|Q|}$ be the vector-row with elements

$$z_o(i) = \begin{cases} 1 & \text{if } q_i=q_o; \\ 0 & \text{if } q_i\neq q_o. \end{cases}$$

The product $z_o.M(u)$ determines the behaviour of A under the input word $u\in X^*$ if the initial state is $q_o\in Q$. Let $z_F=(z_F(i))_{|Q|x1}$ be the column-vector with elements

$$z_F(i) = \begin{cases} 1 & \text{if } q_i\in F, \\ 0 & \text{otherwise.} \end{cases}$$

Then $z_o.M(u).z_f \in [0,1]$ gives the maximal membership degree for the input word $u \in X^*$ if the beginning state is the initial state $q_o \in Q$ and the last state belongs to F. Obviously $u \in L(G_F) \Leftrightarrow z_o.M(u).z_F > 0 \Leftrightarrow u \in R(A)$.

Theorem 5. Let $A=(X,Q,q_o,F,M,\mathbb{L})$ be an acceptor. If $G_F=(N,T,S,P_F)$ is a fuzzy grammar with $N=Q$, $T=X$, $S=q_o$ and productions in P_F have the form $q_i \xrightarrow{p_{ij}} xq_j$ if $m_{ij}(x)=p_{ij}>0$, where $q_i,q_j \in Q$ and $x \in X$, or $q_i \xrightarrow{p_i} x$ if $m_{if}(x)=p_i>0$ for $q_i \in Q$, $q_f \in F$, $x \in X$, then G_F is a grammar of type 3 and $L(G_F)=R(A)$.

We may prove Th.5 in complete analogy with Th.4.

Example 2. Construct the acceptor, corresponding to the fuzzy grammar G_F given in Example 1.

According to Th.4 we obtain $X=\{a,b\}$, $Q=N\cup\{E\}=\{S_o,S_1,S_2,E\}$, $q_o=S_o$, $F=\{E\}$. The transition matrices are:

$$M(a) = \begin{pmatrix} 0 & 0,1 & 0 & 0 \\ 0 & 0,2 & 0 & 0,7 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \qquad M(b) = \begin{pmatrix} 0 & 0 & 0,3 & 0 \\ 0 & 0 & 0,5 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

The direct computation for $M(a^2)$ and $M(b^2)$ is:

$$M(a^2) = \begin{pmatrix} 0 & 0,1 & 0 & 0,1 \\ 0 & 0,2 & 0 & 0,2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \qquad M(b^2) = \begin{pmatrix} 0 & 0 & 0 & 0,3 \\ 0 & 0 & 0 & 0,5 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

By induction we can prove that

$$M(a^n) = \begin{pmatrix} 0 & 0,1 & 0 & 0,1 \\ 0 & 0,2 & 0 & 0,2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, n>1; \quad M(a^n b^2) = \begin{pmatrix} 0 & 0 & 0 & 0,1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \; n \geq 1.$$

Since z_o=(1 0 0 0) and z_F^t = (0 0 0 1) we compute

$$z_o.M(b^2).z_F = 0{,}3;\ z_o.M(a^n).z_F = 0{,}1;\ z_o.M(a^n b^2).z_F = 0{,}1.$$

The other words from X* are not acceptable since $z_o.M(u).z_F=0$. Hence this acceptor recognizes exactly the fuzzy language $L(G_F)$ from Example 1.

3. ε-EQUIVALENCE AND ε-REDUCTION

The classical problems for pure equivalence, reduction and minimization are completely studied for deterministic and non-deterministic automata [5]. Latter they were treated on the lines of their analogues for stochastic [5] and fuzzy [4],[6] automata.

Since the nature of stochastic and fuzzy automata is that they can be thought of as approximate models of incompletely understood systems the idea of approximate equivalence by (stochastic, resp. fuzzy) behaviours is well motivated.

We shall define and study the approximate equivalence and approximate reduction by inputs based on the ε-distance of the behaviour matrices for fuzzy acceptors. The main results concern the algorithmical decidability of the above problems.

The terminology on automata theory is according to [5].

Let $A=(X,Q,q_o,F,M,\mathbb{L})$ be an acceptor. The input words $u,v\in X^*$ are called *ε-equivalent* iff $d(M(u),M(v))\leq\varepsilon$ (notation $u \overset{\varepsilon}{\sim} v$).

Theorem 6. Let $A=(X,Q,q_o,F,M,\mathbb{L})$ be an acceptor, $x,x'\in X$ be input letters and $u,v\in X^*$ be input words. If $x \overset{\varepsilon}{\sim} x'$ then:

i) $uxv \overset{\varepsilon}{\sim} ux'v$;

ii) $xv \overset{\varepsilon}{\sim} x'v$;

iii) $ux \overset{\varepsilon}{\sim} ux'$.

Proof. $x \overset{\varepsilon}{\sim} x' \Longleftrightarrow d(M(x),M(x')) \leqq \varepsilon$. (i) follows from Th.3 (iii), (ii) follows from Th.3(ii);(iii) follows from Th.3(i)

Hence ε-equivalence by input letters implies ε-equivalence by input words, distinguished only by ε-equivalent letters (standing in the middle, in the beginning or in the end of the words).

The relation ε-equivalence by inputs is not an equivalence relation. But we can define an ε-partition on X, resp. on X*.

The set $[x_i]=\{x/x\in X$ and $x \overset{\varepsilon}{\sim} x_i\}$ defines an ε-*class* with *center* x_i. The ε-classes $([x_i])_i$ are called an ε-*partition* of X iff: $[x_i]\cap[x_j]=\emptyset$ for $i\neq j$ and $\bigcup_i [x_i]=X$. According to Th.6 the ε-partition on X induces an ε-partition on X*. Note that $[x_i]\neq[x_j] \Rightarrow d(M(x_i),M(x_j))>\varepsilon$.

Theorem 7. For each acceptor $A=(X,Q,q_o,F,M,\mathbb{L})$ the following problems are algorithmically decidable:

i) whether $x \overset{\varepsilon}{\sim} x'$ for each $x,x'\in X$;

ii) construcing an ε-partition on X (resp. on X*).

Proof. i) For each $x,x'\in X$ we can compute whether $d(M(x), M(x'))\leqq\varepsilon$. The algorithm is finite because A is a finite acceptor. ii) We can construct an ε-partition on X using the following algorithm:

1. Enter X,M,ε.

2. For the elemenet $x_i\in X$ with the smalest index form the ε-class $[x_i]=\{x/x\in X$ and $x \overset{\varepsilon}{\sim} x_i\}$.

3. Print $[x_i]$.

4. $X = X - [x_i]$.

5. If $X\neq\emptyset$ go to Step 2.

6. End.

Let $A=(X,Q,q_o,F,M,\mathbb{L})$ and $A'=(X,Q',q_o',F',M',\mathbb{L})$ be acceptors with the same $X,\mathbb{L}$. A and A' are *ε-equivalent by inputs* ($A \overset{\varepsilon}{\sim} A'$) if for each $x \in X$ there exists an ε-equivalent $x' \in X'$ (i.e. $d(M(x),M'(x')) \leqq \varepsilon$) and vice versa. A' is in *ε-reduced form by inputs* if $x' \overset{\varepsilon}{\sim} x'' \Rightarrow x'=x''$ for each $x',x'' \in X'$. A' is an *ε-reduct* of A if $A \overset{\varepsilon}{\sim} A'$ and A' is in ε-reduced form.

Theorem 8. Let $A \overset{\varepsilon}{\sim} A'$. For each input word $u \in X^*$ there exists an ε-equivalent input word $u' \in X'^*$ and vice versa.

Proof. If $u=e$ or $u \in X$ the proof is trivial. For $u=x_i x_j \in X$ the ε-equivalent word is $u'=x'x'' \in X'^*$ if $x_i \overset{\varepsilon}{\sim} x'$ and $x_j \overset{\varepsilon}{\sim} x''$ because

$$x_i \overset{\varepsilon}{\sim} x' \Rightarrow d(M(x_i),M'(x')) \leqq \varepsilon,$$

$$x_j \overset{\varepsilon}{\sim} x'' \Rightarrow d(M(x_j),M'(x'')) \leqq \varepsilon$$

and according to Th.2 $d(M(x_i),M'(x')) \leqq \varepsilon$ and $d(M(x_j),M'(x'')) \leqq \varepsilon \Rightarrow d(M(x_i x_j),M'(x'x'')) \leqq \varepsilon$. The rest of the proof follows by induction on the lenght of the words and having in mind that $A \overset{\varepsilon}{\sim} A'$.

Corollary. If A' is an ε-reduct of A then A and A' have ε-equivalent behaviours.

Theorem 9. It is algorithmically decidable to find an ε-reduct for each acceptor $A=(X,Q,q_o,F,M,\mathbb{L})$.

Proof. According to Th.7 we can find an ε-partition X_r of X, where X_r is the set of the centers of the ε-classes. The different symbols in X_r are not ε-equivalent by construction. The acceptor $A_r=(X_r,Q,q_o,F,M_r,\mathbb{L})$ with $M_r=\{M(x)/x \in X_r\}$ is an ε-reduct of A.

4. APPLICATIONS IN SYNTACTIC PATTERN RECOGNITION

We shall sektch some applications of these results in syntactic pattern recognition. This is an open problem [1].

Let $W=\{(w,p(w))/p(w)\in[0,1]\}$ be a class of images and $p(w)$ is the membership degree for the image w. If each w is a string we can consider W as a fuzzy language $L(G_F)$. The set of the features P, wich characterizes each string of W, determines a finite set T of the terminals for the fuzzy grammar G_F, respectively the set X of the input letters for the recognizing acceptor A with $R(A)=L(G_F)$.

Let a suitable criterion with a numberical valuation $\varepsilon\in[0,1]$ be choosen, i.e. ε characterizes the similarity measure of the features in W. If the input letters $x,x'\in X$ are ε-equivalent, then $w=uxv$ and $w'=ux'v$ are ε-equivalent. But we can assign to each $x\in X$ a feature $p_x\in P$ and vice versa, i.e. $X\tilde{=}P$; consecuently the feature $p_x\in P$ is a carrier of an ε-equivalent information in comparison with $p_{x'}\in P$. Hence we can consider $p_{x'}$ as an inessential feature for the given recognizing problem or we can interprete $p_{x'}$ as an ε-distored image of p_x. Having in mind Th.7 we can construct an ε-partition of the set of the features P (resp. of W). The elements of the ε-class are ε-equivalent (and ε-distored) in comparison with the center p_x. It follows that p_x can be select as an essential feature (sample) for the recognizing problem. But the choise of the essential features is algorithmically decidable (Th.9) because it is equivalent to the constructing of the ε-reduct A_r for A. With the same notions, if A_r is an ε-reduct of A, then A recognizes images, which are ε-distored in comparison with the images acceptable by A_r.

REFERENCES

1. D.Dubois, H.Prade, "Fuzzy Sets and Systems: Theory and Applications", Acad.Press, New York/London, 1980.

2. K.Fu, "Syntactic Methods in Pattern Recognition", Acad. Press, New York/London, 1974.

3. G.Grätzer, "General Lattice Theory", Birkhaüser Verlag, Basel, 1978

4. E.Santos, On reduction of Maxi-min Machines, J.Math.An. Appl., 40, 1972.

5. P.Starke, Abstracte Automaten, Berlin, 1969.

6. V.Topencharov, K.Peeva, Equivalence, Reduction and Minimization of Finite Fuzzy Automata, j.Math.An.Appl. 84, 1981.

Fuzzy automaták és fuzzy grammatikák

K.G. Peeva

Összefoglaló

Legyen A egy véges fuzzy akceptor és R(A) az A által felismerhető szavak halmaza. A szerző bebizonyitja, hogy minden fuzzy szabályos G_F grammatikához, amely generálja az $L(G_F)$ nyelvet, létezik egy A véges fuzzy akceptor ugy hogy $R(A) = L(G_F)$, és vice versa.

Расплывчатые /fuzzy/ грамматики и множества

К.Г. Пеева

Р е з ю м е

Пусть A есть конечный акцептор и R(A) есть множество всех слов, распознанных акцептором A. Доказывается, что для всех регулярных грамматик G_F ненерирующих язык $L(G_F)$, существует конечный расплывчатый акцептор такой, что $R(A)=L(G_F)$ и наоборот.