Tendências em Matemática Aplicada e Computacional, **19**, N. 1 (2018), 127-145
© 2018 Sociedade Brasileira de Matemática Aplicada e Computacional
www.scielo.br/tema
doi: 10.5540/tema.2018.019.01.0127

Fuzzy Linear Automata and Some Equivalences

VALDIGLEIS S. COSTA[1]* and BENJAMÍN BEDREGAL[2]

Received on March 21, 2017 / Accepted on January 22, 2018

ABSTRACT. In this paper, we present two new normal forms for fuzzy linear grammars. Moreover, we introduce two new classes of fuzzy machines. The first is the class of fuzzy linear automata (FLA). We show that automata of this type is equivalent to the fuzzy linear grammar, i.e., we prove that the class of fuzzy languages is recognized by fuzzy linear automata. Finally, we introduce the class of fuzzy nondeterministic 2-tape automata (FNTA), show the equivalence between them and fuzzy linear automata.

Keywords: Fuzzy grammar, Fuzzy linear grammars, Fuzzy Language, Fuzzy Automata.

1 INTRODUCTION

Formal language theory focuses on the study of sets of words called "languages" and in the algebraic structures that is generated and recognized by these sets of words. As said in [10] some of these algebraic structures that generate these sets are called "grammars". The original Chomsky's hierarchy is composed by four formal languages classes (see [9]) which has been widely and deeply studied. This hierarchy has been extended with some others important classes of formal languages.

The linear languages introduced by Amar and Putzolu [2] is one class of languages that extends the Chomsky's hierarchy. Linear languages are characterized by linear grammars or by states machines presented by Bedregal and Rosemberg, respectively nondeterministic linear automata [5, 4] and nondeterministic 2-tape finite automata [17].

As can be seen in [16] every class of language and respectively your grammar has a fuzzy version. Chaudhari and Komejwar presented in [8] the notions of fuzzy linear grammars and fuzzy linear language. Already in [16, 8, 21] are defined the notions of normal forms for fuzzy regular grammars and fuzzy context-free grammars.

*Corresponding author: Valdigleis S. Costa – E-mail: valdigleis@ppgsc.ufrn.br.

[1] Doutorando em Ciência da Computação, Programa de Pós-graduação em Sistemas e Computação, DIMAp, Universidade Federal do Rio Grande do Norte, Campus Universitário, Lagoa Nova, CEP 59078-970, Caixa postal 1524, Natal/RN – Brasil.

[2] Departamento de Informática e Matemática Aplicada, CCET, Universidade Federal do Rio Grande do Norte, Campus Universitário, Lagoa Nova, CEP 59078-970, Caixa postal 1524, Natal/RN – Brasil. E-mail: bedregal@dimap.ufrn.br

In this paper we will introduce two new normal forms for fuzzy linear grammars. Next we will present a fuzzy generalization for the nondeterministic linear automata [5], that we called fuzzy linear automata (FLA). Then we will show that this new class of fuzzy automata can be used as characterization for fuzzy linear languages. Moreover, we will define the class of the fuzzy 2-tape automata (FNTA), and we will prove that these are equivalent to the fuzzy linear automata, i.e., they are characterization for fuzzy linear languages.

This work will be divided in five sections, this introduction, the Background where is presented the fundamentals required for work, in 3-th section are introduced the new normal forms. In section 4 is presented the news classes of fuzzy states machines, and finally in 5-th section the conclusion of this work.

2 BACKGROUND

In this section, we introduce the basic concepts need for this work.

2.1 Formal Language

The unity basic for study of formal language, grammar and automata is the notion of alphabet.

Definition 1. *An alphabet Σ is an non-empty finite set, the elements of Σ are called letters (or symbols).*

A word on an alphabet Σ is a finite sequence of symbols on Σ. On any alphabet is defined a special word, called empty word, denoted by λ. The empty word is a word without symbols of the alphabet. The size of any word w, denoted by $|w|$, is the number of symbols in w. Thus, if $w = aea$, we have that $|w| = 3$ and for the empty word $|\lambda| = 0$. Some remarkable sets can be obtained from the alphabet Σ.

Definition 2. *For some integer $n \geq 0$ and any alphabet Σ, is defined the set Σ^n of all words of size n.*

Remark 1. *Notice that the set Σ^0 contains only the empty word, i.e., $\Sigma^0 = \{\lambda\}$.*

Definition 3. *For any alphabet Σ is defined, the **positive closure**, denoted by Σ^+ such that, $\Sigma^+ = \bigcup_{i=1}^{\infty} \Sigma^i$ and the **Kleene closure**, denoted by Σ^*, where $\Sigma^* = \Sigma^0 \cup \Sigma^+$.*

Definition 4. *Let $w \in \Sigma^*$ be a word, such that $w = \lambda$ or $w = x_1x_2\ldots x_{n-1}x_n$ with $x_i \in \Sigma$. The functions $R_l, R_r, M_l, M_r : \Sigma^* \to \Sigma^*$ are defined by:*

$$R_l(w) = \begin{cases} x_1, & if\,|w| > 0 \\ \lambda, & else \end{cases} \qquad M_l(w) = \begin{cases} x_2\ldots x_{n-1}x_n, & if\,|w| > 1 \\ \lambda, & else \end{cases}$$

$$R_r(w) = \begin{cases} x_n, & if\,|w| > 0 \\ \lambda, & else \end{cases} \qquad M_r(w) = \begin{cases} x_1x_2\ldots x_{n-1}, & if\,|w| > 1 \\ \lambda, & else \end{cases}$$

Example 1. *For the word "Brazil" we have that $R_l(Brazil) = B$, $M_l(Brazil) = razil$, $R_R(Brazil) = l$ and $R_M(Brazil) = Brazi$.*

Moreover, if $w = a_1a_2\cdots a_{m-1}a_m$ and $w' = b_1b_2\cdots b_{n-1}b_n$, then the concatenation of w with the word w' is the word $ww' = a_1a_2\cdots a_{n-1}a_mb_1b_2\cdots b_{n-1}b_n$. The concatenation of any word w with λ is, $w\lambda = \lambda w = w$. If $w = a_1a_2\cdots a_{m-1}a_m$, then $w^R = a_ma_{m-1}\cdots a_2a_1$ is the reverse of w.

Definition 5. *[6] A language $\mathscr{L}$ is a subset of words on Σ^*, i.e., $\mathscr{L} \subseteq \Sigma^*$.*

2.2 Basic Fuzzy Theory

The fuzzy theory is a generalization introduced by Zadeh in his work [22] of the notion of sets, proposed initially in 1884 by the german mathematician Georg Cantor. The key concept of fuzzy theory is the notion of fuzzy set, where the elements of the universe belongs to the fuzzy set with a membership value, typically a value on [0,1].

Definition 6. *[22] Let $\mathbb{U}$ be a non-empty universe set. A fuzzy set $\boldsymbol{A}$ on $\mathbb{U}$ is characterized by the function $\mu_{\boldsymbol{A}} : \mathbb{U} \to [0,1]$, where $\mu_{\boldsymbol{A}}(x)$ is the membership value of the element $x \in \mathbb{U}$ on fuzzy set $\boldsymbol{A}$.*

Remark 2. *The fuzzy set $\boldsymbol{A}$ can be seen as the set of ordered pairs.*

$$\boldsymbol{A} = \big\{\big(x, \mu_{\boldsymbol{A}}(x)\big) \mid x \in \mathbb{U}\big\}$$

Definition 7. *Let $\mathbb{U}$ be a non-empty universe set. A positive fuzzy semi-set $\boldsymbol{A}$ on $\mathbb{U}$ is characterized by the partial function $\psi_{\boldsymbol{A}} : \mathbb{U} \to]0,1]$.*

It is easy to see that there is at least one bijection from the class of positive fuzzy semi-sets onto the class of fuzzy sets. In [15] Menger introduces the operators t-norms and t-conorms, already Schweizer and Sklar in [18] presented an axiomatic for the t-norms and t-conorms. These operators has been used as standard operators in fuzzy theory studies, see [21, 22, 7, 1, 13].

Definition 8. *[12] A function $\otimes : [0,1]^2 \to [0,1]$ is a t-norm if the following properties are satisfied:*

T_1. *Commutativity: $\otimes(x,y) = \otimes(y,x)$;*

T_2. *Associativity: $\otimes(x,\otimes(y,z)) = \otimes(\otimes(x,y),z)$;*

T_3. *Monotonicity: if $x \leq x'$ and $y \leq y'$, then $\otimes(x,y) \leq \otimes(x',y')$ and*

T_4. *1-neutral: $\otimes(x,1) = x$.*

An element $x \neq 0$ is called zero divisor on a t-norm $\otimes$ if there exist $y \neq 0$ such that, $\otimes(x,y) = 0$. In this paper we will consider only t-norms without zero divisors.

Definition 9. *[12] A function $\oplus : [0,1]^2 \rightarrow [0,1]$ is a t-conorm if the following properties are satisfied:*

S_1. *Commutativity: $\oplus(x,y) = \oplus(y,x)$;*

S_2. *Associativity: $\oplus(x,\oplus(y,z)) = \oplus(\oplus(x,y),z)$;*

S_3. *Monotonicity: if $x \leq x'$ and $y \leq y'$, then $\oplus(x,y) \leq \oplus(x',y')$ and*

S_4. *0-neutral: $\oplus(x,0) = x$.*

As presented t-norms and t-conorm are binary operations, however, it is interesting and convenient extend this notion so that they can receive an n-tuple of input arguments, this extension of the t-norm is defined recursively for some positive integer n as being,

$$\begin{aligned} \bigotimes_{i=1}^{n} x_i &= \otimes\left(\bigotimes_{i=1}^{n-1} x_i, x_n\right) \\ &= \otimes(x_1, x_2, \cdots, x_n) \end{aligned}$$

and

$$\begin{aligned} \bigoplus_{i=1}^{n} x_i &= \oplus\left(\bigoplus_{i=1}^{n-1} x_i, x_n\right) \\ &= \oplus(x_1, x_2, \cdots, x_n) \end{aligned}$$

Remark 3. *In the particular case of $n = 1$ we have that, $\otimes(x_1) = x_1, \oplus(x_1) = x_1$.*

2.3 Fuzzy Languages and Fuzzy Grammars

While in formal languages there is a guarantee of a high level of accuracy, in natural languages there is a high level of imprecision [16]. In an attempt to narrow the gap between formal languages and natural languages Lee and Zadeh introduced fuzzy languages and fuzzy grammars [13].

Definition 10. *[13] A fuzzy language $\mathbb{L}$ is a fuzzy set on Σ^*. Thus $\mathbb{L}$ is a set of ordered pairs $\mathbb{L} = \left\{\left(w, \mu_{\mathbb{L}}(w)\right) \mid w \in \Sigma^*\right\}$.*

As said in [16], fuzzy grammars are sets of rules capable of generating the elements of a fuzzy set, thus, are generators of fuzzy languages.

Definition 11. *[8] A Fuzzy Context-Free Grammar (FCG) is a structure $\mathscr{G} = \langle V, \Sigma, \hat{S}, \hat{P}, \otimes, \oplus \rangle$ where:*

V is a finite set of variables and Σ is an alphabet[1]*;*

[1] We emphasize that V and Σ are disjoint sets.

$\hat{S} : V \to [0,1]$ *is a non-empty fuzzy set;*

$\hat{P} : V \times (V \cup \Sigma)^* \to]0,1]$ *is a positive fuzzy semi-set of the productions.*

$\otimes$ *is a t-norm and* $\oplus$ *is a t-conorm.*

As in [16, 8, 13] for simplicity, $\hat{P}(\alpha,\beta) = \sigma$ is written as $\alpha \xrightarrow{\sigma} \beta$ for some $\sigma \in [0,1], \alpha \in V$ and $\beta \in (V \cup \Sigma)^*$, in this case we denote $\alpha \xrightarrow{\sigma} \beta \in \hat{P}$.

Example 2. *The structure* $\mathscr{G} = \langle \{A,B,X\}, \{a,b\}, \{(X,0.9)\}, \hat{P}, min, max \rangle$ *is a FCG where* $\hat{P} = \{X \xrightarrow{0.8} A, X \xrightarrow{0.8} B, A \xrightarrow{0.5} a, A \xrightarrow{0.6} b, B \xrightarrow{0.4} A, B \xrightarrow{0.2} a\}$.

Definition 12. *Given a fuzzy grammar* $\mathscr{G}$*, for all* $w, w' \in (V \cup \Sigma)^*$ *we define the set,*

$$\Gamma_{w,w'} = \left\{ \sigma \mid (\exists x, y, \alpha, \beta \in \Sigma^*), w = x\alpha y, w' = x\beta y \text{ and } \alpha \xrightarrow{\sigma} \beta \in \hat{P} \right\}$$

As in [16] we say that $w' \in (V \cup \Sigma)^*$ is **directly derived** from $w \in (V \cup \Sigma)^*$ in a FCG $\mathscr{G}$, denoted by relation $w \models_{\mathscr{G}} w'$, if $\Gamma_{w,w'} \neq \emptyset$. There are several forms of calculating the membership value of direct derivation from one word to another [8, 21, 13, 3], and here we present our interpretation of how to obtain such a value.

Definition 13. *Let* $\mathscr{G}$ *be a fuzzy grammar, the membership value of* $w \models_{\mathscr{G}} w'$ *is given by the function* $d_{\mathscr{G}} : (V \cup \Sigma)^* \times (V \cup \Sigma)^* \to [0,1]$ *where* $d_{\mathscr{G}}$ *is defined as:*

$$d_{\mathscr{G}}(w,w') = \bigvee \Gamma_{w,w'} \tag{2.1}$$

with $\bigvee$ *being the operation of supremum in the poset* $\langle [0,1], \leq \rangle$.

If $w_1, w_2, \cdots, w_{n-1}, w_n \in (V \cup \Sigma)^*$ and $w_1 \models_{\mathscr{G}} w_2, w_2 \models_{\mathscr{G}} w_3, \cdots, w_{n-1} \models_{\mathscr{G}} w_n$, then we say that there exist a derivation from w_1 onto w_n. We defined the binary relation $\models_{\mathscr{G}}^*$ as being the reflexive and transitive closure of $\models_{\mathscr{G}}$ and it's membership value is defined as follows.

Definition 14. *Let* $\mathscr{G}$ *be a fuzzy grammar, the membership value of* $w \models_{\mathscr{G}}^* w'$ *is given by the function* $\hat{d}_{\mathscr{G}} : (V \cup \Sigma)^+ \times (V \cup \Sigma)^* \to [0,1]$ *defined by:*

$$\hat{d}_{\mathscr{G}}(w,w') = \begin{cases} 1, & \text{if } w = w' \\ \bigoplus_{w'' \in (V \cup \Sigma)^+} \left(\otimes(\hat{d}_{\mathscr{G}}(w,w''), d_{\mathscr{G}}(w'',w')) \right), & \text{else} \end{cases} \tag{2.2}$$

Example 3. *Let G be the fuzzy grammar of Example 2. The membership value of* $X \models_{\mathscr{G}}^* a$ *is obtained by:*

$$\begin{aligned}
\hat{d}(X,a) &= max\left(min(\hat{d}(X,A), d(A,a)), min(\hat{d}(X,B), d(B,a))\right) \\
&= max\left(min\left(max\left(min(1,0.8), min(\hat{d}(X,B),0.4)\right), 0.5\right), min(\hat{d}(X,B),0.2)\right) \\
&= max\Big(min\Big(max\,(0.8, min(0.8,0.4)), 0.5\Big), min(0.8,0.2)\Big) \\
&= max\,(min\,(0.8,0.5), 0.2) \\
&= 0.5
\end{aligned}$$

The membership value of $w \in \Sigma^*$ be generated by a FCG $\mathscr{G}$, is the value of all derivations from α onto w, where $\alpha \in V$, such membership value is given by the equation,

$$\mu_{\mathscr{G}}(w) = \bigoplus_{\alpha \in V} \left(\otimes(\hat{S}(\alpha), \hat{d}(\alpha, w)) \right) \tag{2.3}$$

Definition 15. *The fuzzy language of a fuzzy grammar $\mathscr{G}$ is the fuzzy set of all words generated by G, defined by, $L(\mathscr{G}) = \{(w, \mu_{\mathscr{G}}(w)) \mid w \in \Sigma^*\}$.*

Definition 16. *Let $\mathscr{G}$ be a FCG, the productions of the form $A \xrightarrow{\sigma} C \in \hat{P}$ with $A, C \in V$ are called unitary productions.*

Lemma 1. *If $\mathscr{G}$ is a FCG, then there is a FCG $\mathscr{G}'$ without unitary productions such that $L(\mathscr{G}) = L(\mathscr{G}')$.*

Proof. See the first step of the algorithm for obtaining Chomsky's normal form in page 193 from [16].

Fuzzy languages as the "classical" languages can be classified according to their type of generating grammar [16] in classes or families, this organization constitutes a fuzzy version of Chomsky's hierarchy[9]. Besides the types of grammar presented in [16], there exist another type of fuzzy grammar, called fuzzy linear grammar [8].

Definition 17. *[8] A FCG $\mathscr{G}$ will be said be a Fuzzy Linear Grammar (FLG) if for all $\alpha \xrightarrow{\sigma} \beta \in \hat{P}$, we have $\beta = x_1 A x_2$ and $A \in (V \cup \{\lambda\})$ and $x_1, x_2 \in \Sigma^*$.*

Remark 4. *Notice that every FLG is a FCG, but the opposite is not true.*

Definition 18. *A fuzzy language $\mathbb{L}$ is a fuzzy linear language, if there exists a FLG $\mathscr{G}$ such that $L(\mathscr{G}) = \mathbb{L}$.*

3 NEW NORMAL FORMS

In [16, 8] we find a notion of normal form for fuzzy grammars, however, this normal form was directed only to the class of grammars type 0 (or fuzzy regular grammars). In [21] we have the idea of normal form for fuzzy free context grammars. We now propose two new normal forms for the class of fuzzy linear grammars.

Definition 19. *Let $\mathscr{G}$ be a FLG, then for all $\alpha \in V$ we say:*

1. *α is **left-linear** (or LL), if for all $\alpha \xrightarrow{\sigma} \beta \in \hat{P}$, we have $\beta = Ba$, where $a \in \Sigma^+$ and $B \in V$.*
2. *α is **right-linear** (or RL), if for all $\alpha \xrightarrow{\sigma} \beta \in \hat{P}$, we have $\beta = aB$, where $a \in \Sigma^+$ and $B \in V$.*
3. *α is **without direction** (or WD), if for all $\alpha \xrightarrow{\sigma} \beta \in \hat{P}$, we have $\beta \in \Sigma^*$ or $\beta \in V$.*

Definition 20. *Let $\mathscr{G}$ be a FLG we define the sets, $V_L^{\mathscr{G}} = \{\alpha \in V \mid \alpha \text{ is LL}\}$ and $V_R^{\mathscr{G}} = \{\alpha \in V \mid \alpha \text{ is RL}\}$.*

Lemma 2. *For all FLG $\mathscr{G}$, there exist a FLG $\mathscr{G}'$ in LNF without production of the form $A \xrightarrow{\sigma} uBv$ with $A, B \in V$ and $u, v \in \Sigma^+$ such that $L(\mathscr{G}) = L(\mathscr{G}')$.*

Proof. (Sketch) Without loss of generality assume $\mathscr{G} = \langle V, \Sigma, \hat{S}, \hat{P}, \otimes, \oplus \rangle$ does not contain unitary productions (Lemma 1). Thus we build $\mathscr{G}' = \langle V', \Sigma, \hat{S}, \hat{P}', \otimes, \oplus \rangle$ by the following algorithm.

1. Initially do $V' = V$ and $\hat{P}' = \hat{P}$.
2. For all $A \in V$ do:

   2.1 If $A_u = \{A \xrightarrow{\sigma} uBv \in \hat{P} \mid \text{for some } B \in V \text{ and } u, v \in \Sigma^+\} \neq \emptyset$, then for each production $A \xrightarrow{\sigma} uBv \in A_u$ define in $\hat{P}'$ the production $A \xrightarrow{1} uC$ and $C \xrightarrow{\sigma} Bv$ in $\hat{P}'$, where C is a new variable add in V', finally remove $A \xrightarrow{\sigma} uBv$ of the $\hat{P}'$.

To show that $L(\mathscr{G}) = L(\mathscr{G}')$ we must show that for all production in $\mathscr{G}$ there exist a derivation equivalent in $\mathscr{G}'$. Suppose that in $\mathscr{G}$ there exist $A \xrightarrow{\sigma} uBv \in \hat{P}$, therefore by definition:

$$\begin{aligned}
\hat{d}_{\mathscr{G}}(A, uBv) &= \bigoplus \left(\otimes(\hat{d}_{\mathscr{G}}(A, A), d_{\mathscr{G}}(A, uBv))\right) \\
&= \bigoplus (\otimes(1, \sigma)) \\
&= \sigma \\
&= d_{\mathscr{G}}(A, uBv)
\end{aligned}$$

But by step 2.1 exist in $\hat{P}'$ productions $A \xrightarrow{1} uC$ and $C \xrightarrow{\sigma} Bv$, so in $\mathscr{G}'$ we have,

$$\begin{aligned}
\hat{d}_{\mathscr{G}'}(A, uBv) &= \bigoplus \Big(\otimes \left(\hat{d}_{\mathscr{G}'}(A, uC), d_{\mathscr{G}'}(uC, uBv)\right)\Big) \\
&= \bigoplus \Big(\otimes (\bigoplus \Big(\otimes \left(\hat{d}_{\mathscr{G}'}(A, A), d_{\mathscr{G}'}(A, uC)\right)\Big), d_{\mathscr{G}'}(uC, uBv))\Big) \\
&= \bigoplus \Big(\otimes (\bigoplus \Big(\otimes (1, 1)\Big), \sigma)\Big) \\
&= \bigoplus \Big(\otimes (1, \sigma)\Big) \\
&= \sigma \\
&= d_{\mathscr{G}}(A, uBv) \\
&= \hat{d}_{\mathscr{G}}(A, uBv).
\end{aligned}$$

Thus, for all derivation that use production of the form $A \xrightarrow{\sigma} uBv$ in $\mathscr{G}$ there exist in $\mathscr{G}'$ a derivation equivalent that use the productions $A \xrightarrow{1} uC$ and $C \xrightarrow{\sigma} Bv$. Therefore, we can conclude that the algorithm preserves degrees membership, so for all $x \in \Sigma^*$ we have to that $\mu_{\mathscr{G}}(x) = \mu_{\mathscr{G}'}(x)$, implying that $L(\mathscr{G}) = L(\mathscr{G}')$.

A We said that a FLG $\mathscr{G}$ will be in **linear normal form** (LNF) only when $V_L^{\mathscr{G}} \cap V_R^{\mathscr{G}} = \emptyset$. Notice that, unlike the normal form in [8], the LNF does not require grammar to be strict in a direction.

Theorem 1. *For all FLG $\mathscr{G}$, there exist a FLG $\mathscr{G}'$ in LNF such that $L(\mathscr{G}) = L(\mathscr{G}')$.*

Proof. (Sketch) Without loss of generality assume that $\mathscr{G} = \langle V, \Sigma, \hat{S}, \hat{P}, \otimes, \oplus \rangle$ does not contain productions $A \xrightarrow{\sigma} uBv$ (Lemma 2), so all production in $\hat{P}$ are of forms LL, RL and WR, thus $\mathscr{G} = \langle V', \Sigma, \hat{S}, \hat{P}', \otimes, \oplus \rangle$ where $\hat{P}'$ is build by the following algorithm:

1. Initially do $V' = V$ and $\hat{P}' = \hat{P}$.

2. For all $A \in V$ do:

   2.1 If $A \in V_L^{\mathscr{G}} \cap V_R^{\mathscr{G}}$, then for each $A \xrightarrow{\sigma'} Bu \in \hat{P}'$ add the productions $A \xrightarrow{1} C$ and $C \xrightarrow{\sigma} Bv$ in $\hat{P}$ such that C is a new variable. Add C in V', finally remove $A \xrightarrow{\sigma} Bv$ of the $\hat{P}'$.

Is easy see that the algorithm guarantees that for all $A \in V$ if $A \in V_L^{\mathscr{G}'}$, then $A \notin V_R^{\mathscr{G}'}$, so after executing the algorithm we will have that $V_L^{\mathscr{G}'} \cap V_L^{\mathscr{G}'} = \emptyset$. Now notice that, if $A \xrightarrow{\sigma} Bv \in \hat{P}$ by definition,

$$\begin{array}{rcl} \hat{d}_{\mathscr{G}}(A,Bv) & = & \bigoplus \left(\otimes(\hat{d}_{\mathscr{G}}(A,A), d_{\mathscr{G}}(A,Bv))\right) \\ & = & \bigoplus(\otimes(1,\sigma)) \\ & = & \sigma. \end{array}$$

but by algorithm from construction of $\mathscr{G}'$ (step 2.1) we have $A \xrightarrow{1} C, C \xrightarrow{\sigma} Bu \in \hat{P}'$ and by definition,

$$\begin{array}{rcl} \hat{d}_{\mathscr{G}'}(A,Bv) & = & \bigoplus \left(\otimes(\hat{d}_{\mathscr{G}'}(A,C), d_{\mathscr{G}'}(C,Bv))\right) \\ & = & \bigoplus \Big(\otimes(\bigoplus \Big(\otimes\left(\hat{d}_{\mathscr{G}'}(A,A), d_{\mathscr{G}'}(A,C)\right)\Big), d(C,Bv))\Big) \\ & = & \bigoplus \Big(\otimes(\bigoplus \Big(\otimes(1,1)\Big), \sigma)\Big) \\ & = & \bigoplus \Big(\otimes(1,\sigma)\Big) \\ & = & \sigma \\ & = & \hat{d}_{\mathscr{G}}(A,Bv). \end{array}$$

Thus, for all $A \in V_L^{\mathscr{G}} \cap V_R^{\mathscr{G}}$ we have that for all yours productions of the form $A \xrightarrow{\sigma} Bv$, there exist in $\hat{P}'$ the productions $A \xrightarrow{1} C$ and $C \xrightarrow{\sigma} Bu$, so for all derivation that use productions of the form $A \xrightarrow{\sigma} Bv$ there exist in $\mathscr{G}'$ a derivation equivalent that use the productions $A \xrightarrow{1} C$ and $C \xrightarrow{\sigma} Bu$. Therefore, we can conclude that the algorithm preserves degrees membership, so for all $x \in \Sigma^*$ we have to that $\mu_{\mathscr{G}}(x) = \mu_{\mathscr{G}'}(x)$, implying that $L(\mathscr{G}) = L(\mathscr{G}')$.

A FLG will be in **stronger linear normal form** (SLNF) if all productions of $\hat{P}$ are the form $A \xrightarrow{\sigma} aB$ or $A \xrightarrow{\sigma} Ba$ with $A \in V$, $B \in V \cup \{\lambda\}, a \in \Sigma \cup \{\lambda\}$ and $\sigma > 0$.

Theorem 2. *For all fuzzy linear grammar $\mathscr{G}$. There is a linear fuzzy grammar fuzzy $\mathscr{G}'$ in SLNF such that $L(\mathscr{G}) = L(\mathscr{G}')$.*

Proof. (Sketch) Without loss of generality assume $\mathscr{G} = \langle V, \Sigma, \hat{S}, \hat{P}, \otimes, \oplus \rangle$ is in form LNF. We build then $\mathscr{G}' = \langle V', \Sigma, \hat{S}, \hat{P}', \otimes, \oplus \rangle$ using the algorithm:

1. Initially do $V' = V$ and $\hat{P}' = \hat{P}$.
2. For all $A \in V'$ and $a \in \Sigma$, if $A^a = \{A \xrightarrow{\sigma} ya \in \hat{P} \mid y \in \Sigma \text{ or } |y| \geq 2\} \neq \emptyset$, then add a new variable C in V' and for all productions $A \xrightarrow{\sigma} ya \in A^a$ define in $\hat{P}'$ the productions $A \xrightarrow{1} Ca$ and $C \xrightarrow{\sigma} y$ and remove $A \xrightarrow{\sigma} ya$ from $\hat{P}'$.
3. For all $A \in V'$ and $a \in \Sigma$, if $A_a = \{A \xrightarrow{\sigma} ay \in \hat{P} \mid y \in \Sigma \text{ or } |y| \geq 2\} \neq \emptyset$, then add a new variable C in V' and for all productions $A \xrightarrow{\sigma} ay \in A_a$ define in $\hat{P}'$ the productions $A \xrightarrow{1} Ca$ and $C \xrightarrow{\sigma} y$ and remove $A \xrightarrow{\sigma} ay$ from $\hat{P}'$.

To show that $L(\mathscr{G}) = L(\mathscr{G}')$ as before, we must show that the algorithm does not change the membership value of derivations. Let's look at the case of step 2 (the same is similar to step 3). Suppose that $A \in V$ and $A \xrightarrow{\sigma} ya$ with $|y| \geq 2$ or $y \in \Sigma$ and $\sigma > 0$, so we have,

$$\begin{aligned} \hat{d}_{\mathscr{G}}(A, ya) &= \bigoplus \Big(\otimes \big(\hat{d}_{\mathscr{G}}(A, A), d_{\mathscr{G}}(A, ya) \big) \Big) \\ &= \bigoplus \Big(\otimes (1, \sigma) \Big) \\ &= \sigma. \end{aligned}$$

By construction of the $\mathscr{G}'$ exist the productions $A \xrightarrow{1} Ca$ and $C \xrightarrow{\sigma} y$. So,

$$\begin{aligned} \hat{d}_{\mathscr{G}'}(A, ay) &= \bigoplus \big(\otimes (\hat{d}_{\mathscr{G}'}(A, Ca), d_{\mathscr{G}'}(Ca, ya)) \big) \\ &= \bigoplus (\otimes (1, \sigma)) \\ &= \sigma \\ &= \hat{d}_{\mathscr{G}}(A, ya). \end{aligned}$$

So the algorithm preserved membership value of derivations, since for every production $A \xrightarrow{\sigma} ya$ the algorithm constructs the productions $A \xrightarrow{1} Ca$ and $C \xrightarrow{\sigma} y$, thus is easy to see that if in a derivation in $\mathscr{G}$ that use the production $A \xrightarrow{\sigma} ya$, then there exist in $\mathscr{G}'$ an derivation equivalent that use the productions $A \xrightarrow{1} Ca$ and $C \xrightarrow{\sigma} y$. Therefore, for all $x \in \Sigma^*$ we have, $\mu_{\mathscr{G}}(x) = \mu_{\mathscr{G}'}(x)$, so $L(\mathscr{G}) = L(\mathscr{G}')$.

4 TWO NEW FUZZY MACHINES

The interest in the study of fuzzy computing machines is related to both practical application of such models [16, 8, 14] and by the theoretical aspects [21, 7, 20, 19]. In this section, we will introduce two new classes of fuzzy automata, the fuzzy linear automata and the fuzzy nondeterministic 2-tapes automata. These models are respectively adaptations for the fuzzy domain of the model λ-ALN introduced by Bedregal in [5] and of model 2-tape Automata introduced in [17].

4.1 Fuzzy Linear Automata

Definition 21. *Fuzzy Linear Automata(FLA) are structures* $\mathscr{M} = \langle Q_L, Q_R, \Sigma, \phi, \hat{I},\ F, \otimes, \oplus\rangle$ *where:*

Q_L *and* Q_R *are disjoint finite sets of states;*

Σ *is a alphabet;*

$\phi : (Q_L \cup Q_R) \times (\Sigma \cup \{\lambda\}) \times (Q_L \cup Q_R) \to]0,1]$ *is a positive fuzzy semi-set of instructions;*

$\hat{I} : (Q_L \cup Q_R) \to [0,1]$ *is a non-empty fuzzy set of initial states;*

$F \subseteq (Q_L \cup Q_R)$ *is a set of final states;*

$\otimes$ *is a t-norm and* $\oplus$ *is a t-conorm.*

To facilitate our presentation, we will write $\phi(q,a,q') = \sigma$ as $(q,q')_a = \sigma$, in this case we denote $(q,q')_a \in \phi$. The FLA's have the same characteristics in relation to "architecture" as the automata in [5, 4], i.e., a FLA is a machine with a memory (tape) read and two heads for reading of tape. Working with FLA's we can adopt the idea of **instant description** (ID). A ID is a pair (q,w) where $q \in (Q_L \cup Q_R)$ and $w \in \Sigma^*$.

Definition 22. *Let* $\mathscr{M}$ *be a FLA, for all ID* (q,w) *we define the sets*

i. $\lambda_{suc}^{(q,w)} = \{(q',w) \mid q' \in (Q_L \cup Q_R)$ *and* $(q,q')_\lambda \in \phi\}$.

ii. $L_{suc}^{(q,w)} = \{(q',w') \mid q' \in (Q_L \cup Q_R), w' = M_l(w)$ *and* $(q,q')_{R_l(w)} \in \phi\}$.

iii. $R_{suc}^{(q,w)} = \{(q',w') \mid q' \in (Q_L \cup Q_R), w' = M_r(w)$ *and* $(q,q')_{R_r(w)} \in \phi\}$.

iv. $SUC^{(q,w)} = \lambda_{suc}^{(q,w)} \cup L_{suc}^{(q,w)} \cup R_{suc}^{(q,w)}$.

For all (q,w) and (q',w') the relation $(q,w) \succ_{\mathscr{M}} (q',w')$ denotes the relation of computation, from (q,w) onto (q',w') on FLA $\mathscr{M}$. The membership value of computation of (q,w) onto (q',w') defined below.

Definition 23. *We defined the fuzzy set* $D_{\mathscr{M}} : (Q_L \cup Q_R) \times \Sigma^* \times (Q_L \cup Q_R) \times \Sigma^* \to [0,1]$ *as being,*

$$D_{\mathscr{M}}(q,w,q',w') = \begin{cases} (q,q')_\lambda, & \textit{if } (q',w') \in \lambda_{suc}^{(q,w)} \\ (q,q')_{R_l(w)}, & \textit{if } q \in Q_L \textit{ and } (q',w') \in L_{suc}^{(q,w)} \\ (q,q')_{R_r(w)}, & \textit{if } q \in Q_R \textit{ and } (q',w') \in R_{suc}^{(q,w)} \\ 0, & \textit{any other case.} \end{cases} \tag{4.1}$$

Example 4. *Let $\mathscr{M} = \langle\{q_0,q_1\},\{p_0\},\{a,b,c\},\mu,\{(q_0,0.9),(q_1,1)\},\{q_0,q_1\},min,max\rangle$ be a FLA, where $(q_0,p_0)_a = 0.9$, $(q_0,q_1)_b = 1$ and $(p_0,q_0)_c = 0.7$. Thus, the membership value of $(q_0,abaa) \succ_{\mathscr{M}} (p_0,baa)$ is*

$$\begin{array}{rcl} D_{\mathscr{M}}(q_0,abaa,p_0,baa) & = & (q_0,p_0)_{R_l(abaa)} \\ & = & (q_0,p_0)_a \\ & = & 0.9 \end{array}$$

The relation binary $\succ^*_{\mathscr{M}}$ is the reflexive and transitive closure of $\succ_{\mathscr{M}}$ and we extend the $D_{\mathscr{M}}$ function by the function $\overline{D}_{\mathscr{M}}$ below.

Definition 24. *The membership value of $(q,w) \succ^*_{\mathscr{M}} (q'',w'')$ is given by the function $\overline{D}_{\mathscr{M}} : (Q_L \cup Q_R) \times \Sigma^* \times (Q_L \cup Q_R) \times \Sigma^* \to [0,1]$ define as being,*

$$\overline{D}_{\mathscr{M}}(q,w,q'',w'') =$$

$$\begin{cases} D(q,w,q'',w''), & \text{if } (q'',w'') \in SUC^{(q,w)} \\ \bigoplus_{(q',w') \in SUC^{(q,w)}} \left(\otimes(D_{\mathscr{M}}(q,w,q',w'),\overline{D}_{\mathscr{M}}(q',w',q'',w''))\right), & \text{if } SUC^{(q,w)} \neq \emptyset \\ 0, & \text{any other case.} \end{cases}$$

Example 5. *Consider the FLA of Example 4 the membership value of chain of computation $(q_0,abc) \succ^*_{\mathscr{M}} (p_0,\lambda)$ is given by:*

$$\begin{array}{rcl} \overline{D}_{\mathscr{M}}(q_0,abc,p_0,\lambda) & = & max\Big(min(D_{\mathscr{M}}(q_0,abc,p_0,bc),\overline{D}_{\mathscr{M}}(p_0,bc,p_0,\lambda))\Big) \\ & = & max\Big(min(0.9,max\Big(min(0.7,1)\Big))\Big) \\ & = & max\,(min(0.9,0.7)) \\ & = & 0.7 \end{array}$$

It is known that an automaton recognizes an input word if from one initial state he consumes all the symbols of the input word and arrives in a final state [6, 11]. We can then define this same concept for FLA's as follows.

Definition 25. *Let $\mathscr{M} = \langle Q_L,Q_R,\Sigma,\phi,\hat{I},F,\otimes,\oplus\rangle$ be a FLA and a word $w \in \Sigma^*$. Then the membership value that $\mathscr{M}$ accept w is:*

$$deg_{\mathscr{M}}(w) = \bigoplus_{q \in (Q_L \cup Q_R), q' \in F} \left(\otimes(\hat{I}(q),\overline{D}_{\mathscr{M}}(q,w,q',\lambda))\right) \tag{4.2}$$

Definition 26. *The language of any FLA $\mathscr{M}$ is the fuzzy set of words accepted by $\mathscr{M}$, i.e., $L(\mathscr{M}) = \{(w,deg_{\mathscr{M}}(w)) \mid w \in \Sigma^*\}$.*

Now we present two results that show the direct relationship between FLA's and fuzzy linear grammars.

Theorem 3. *Let $\mathscr{M}$ be a FLA. Then there exist a fuzzy linear grammar $\mathscr{G}$ such that $L(\mathscr{G}) = L(\mathscr{M})$.*

Proof. First given $\mathscr{M} = \langle Q_L, Q_R, \Sigma, \phi, \hat{I}, F, \otimes, \oplus \rangle$ we build the grammar $\mathscr{G} = \langle Q_L \cup Q_R, \Sigma, \hat{I}, \hat{P}, \otimes, \oplus \rangle$ where

$$\begin{aligned} \hat{P} \;=\; & \{q' \xrightarrow{\sigma} aq'' \mid (q', q'')_a = \sigma, q' \in Q_L, q'' \in (Q_L \cup Q_R), a \in \Sigma \cup \{\lambda\}\} \cup \\ & \{q' \xrightarrow{\sigma'} q''a \mid (q', q'')_a = \sigma', q' \in Q_R, q'' \in (Q_L \cup Q_R), a \in \Sigma \cup \{\lambda\}\} \cup \\ & \{q' \xrightarrow{1} \lambda \mid q' \in F\}. \end{aligned}$$

Clearly $\mathscr{G}$ is a fuzzy linear grammar. Now for all $(w, \sigma) \in L(\mathscr{M})$ where the word w is $a_1 a_2 \cdots a_{n-1} a_n$, such that $a_i \in (\Sigma \cup \{\lambda\})$, imply what exist $q_0, q_n \in (Q_L \cup Q_R)$ of form what $(q_0, w) \succ^*_{\mathscr{M}} (q_n, \lambda)$ with $\hat{I}(q_0) > 0, q_n \in F$. This imply that $deg_{\mathscr{M}}(w) = \bigoplus\{\otimes(\hat{I}(q_0), \hat{D}(q_0, w', q_n, \lambda))\} = \sigma$. However from $(q_0, w) \succ^*_{\mathscr{M}} (q_n, \lambda)$ we have that there exist fuzzy instructions:

$$(q_0, q_1)_{a_1} = \sigma_1, (q_1, q_2)_{a_2} = \sigma_2, \cdots, (q_{n-2}, q_{n-1})_{a_{n-1}} = \sigma_{n-1}, (q_{n-1}, q_n)_{a_n} = \sigma_n$$

Therefore, for each $(q_i, q_{i+1})_{a_{i+1}}$ with $0 \leq i \leq n-1$ by construction of $\hat{P}$ there exist a corresponding fuzzy production, which will be of the form $q_i \xrightarrow{\sigma_{i+1}} a_{i+1} q_{i+1}$ whenever $q_i \in Q_L$, or of form $q_i \xrightarrow{\sigma_{i+1}} q_{i+1} a_{i+1}$ whenever $q_i \in Q_R$ and also a fuzzy production $q_n \xrightarrow{1} \lambda$ for all $q_n \in F$. Now consider $w = y_1 y_2$ for some $y_1, y_2 \in \Sigma^*$, so by construction of $\mathscr{G}$ there exist a variable $q_0 \in V$ and a word $w' = y_1 q_n y_2$, such that the derivation $\hat{d}_{\mathscr{G}}(q_0, w')$ using all productions correspond the instructions:

$$(q_0, q_1)_{a_1} = \sigma_1, (q_1, q_2)_{a_2} = \sigma_2, \cdots, (q_{n-2}, q_{n-1})_{a_{n-1}} = \sigma_{n-1}, (q_{n-1}, q_n)_{a_n} = \sigma_n$$

and by $q_n \in F$ there a production $q_n \xrightarrow{1} \lambda$, thus, $y_1 q_n y_2 \models_{\mathscr{G}} y_1 y_2$, and this implies,

$$\begin{aligned} \hat{d}_{\mathscr{G}}(q_0, w) &= \bigoplus \left(\otimes(\hat{d}_{\mathscr{G}}(q_0, w'), d_{\mathscr{G}}(w', w))\right) \\ &= \bigoplus \left(\otimes(\hat{d}_{\mathscr{G}}(q_0, y_1 q_n y_2), d_{\mathscr{G}}(y_1 q_n y_2, y_1 y_2))\right) \\ &= \bigoplus \left(\otimes(\hat{d}_{\mathscr{G}}(q_0, y_1 q_n y_2), 1)\right) \\ &= \bigoplus \left(\hat{d}_{\mathscr{G}}(q_0, w')\right) \end{aligned}$$

and thus, $\mu_{\mathscr{G}}(w) = \bigoplus \left(\otimes(\hat{I}(q_0), \hat{d}_{\mathscr{G}}(q_0, w'))\right)$ and by construction of $\mathscr{G}$ we have $\hat{d}(q_0, w')$ has a fuzzy production corresponding to each fuzzy instruction $(q_i, q_{i+1})_{a_{i+1}}$ used in $\overline{D}_{\mathscr{M}}(q_0, w, q_n, \lambda)$, imply $\hat{d}(q_0, w') = \overline{D}_{\mathscr{M}}(q_0, w, q_n, \lambda)$, therefore,

$$\mu_{\mathscr{G}}(w) = \bigoplus \left(\otimes(\hat{I}(q_0), \overline{D}_{\mathscr{M}}(q_0, w', q_n, \lambda))\right).$$

Moreover, by definition $deg_{\mathscr{M}}(x) = \bigoplus \left(\otimes(\hat{I}(q_0), \overline{D}_{\mathscr{M}}(q_0, w', q_n, \lambda))\right)$, and thus, $\mu_{\mathscr{G}}(w) = deg_{\mathscr{M}}(w)$, so $L(\mathscr{G}) = L(\mathscr{M})$.

Theorem 4. *Let $\mathscr{G}$ be a fuzzy linear grammar. Then there exist a FLA $\mathscr{M}$ such that $L(\mathscr{M}) = L(\mathscr{G})$.*

Proof. Without loss of generality assume that $\mathscr{G} = \langle V, \Sigma, \hat{S}, \hat{P}, \otimes, \oplus \rangle$ is in the SLNF, thus build the FLA $\mathscr{M} = \langle Q_L, Q_R, \Sigma, \phi, \hat{S}, F, \otimes, \oplus \rangle$ where:

(i) $Q_L = \{A|A \text{ is a variable LR}\} \cup \{C\}$, $Q_R = \{A| \text{ is a variable LL}\}$ and C is a new label such that $C \notin V$, and finally, $F = \{C\}$.

(ii) ϕ is defined for each $A, B \in V$ and $a \in (\Sigma \cup \{\lambda\})$ as:

$$\begin{aligned} (A,B)_a &= \sigma, \text{if } A \xrightarrow{\sigma} aB \text{ or } A \xrightarrow{\sigma} Ba \in \hat{P} \\ (A,C)_a &= \sigma, \text{if } A \xrightarrow{\sigma} a \in \hat{P}. \end{aligned}$$

Clearly $\mathscr{M}$ is a FLA. Now for all $(w, \sigma) \in L(\mathscr{G})$ with $w = a_1 a_2 \cdots a_{n-1} a_n$ such that $a_i \in (\Sigma \cup \{\lambda\})$, we have $\mu_{\mathscr{G}}(w) = \sigma$. Thus, there exist a $A_1 \in V$ such that $\hat{d}_{\mathscr{G}}(A_1, w) = \sigma$, but of this, we have that the i-th production used in derivation is of the form $A_i \xrightarrow{\sigma_i} a_i A_{i+1}$(or $A_i \xrightarrow{\sigma_i} a_i A_{i+1}$) with $A_i \in V, A_{i+1} \in (V \cup \{\lambda\}), a_i \in (\Sigma \cup \{\lambda\})$ and $\sigma_i \in]0,1]$, and last production used is $A_n \xrightarrow{\sigma_n} a_n$ where $A_n \in V, a_n \in (\Sigma \cup \{\lambda\})$. By construction from ϕ exist fuzzy instructions corresponding to each fuzzy production, the fuzzy instruction will be of the form:

$$(A_1, A_2)_{a_1} = \sigma_1, (A_2, A_3)_{a_2} = \sigma_2, \cdots, (A_{n-1}, A_n)_{a_{n-1}} = \sigma_{n-1}, (A_n, C)_{a_n} = \sigma_n$$

and by construction from Q_L and Q_R we have $C \in F$, so it's easy to see that $(A, w) \succ^*_{\mathscr{M}} (C, \lambda)$, therefore,

$$deg_{\mathscr{M}}(w) = \bigoplus \left(\otimes(\hat{S}(A), \overline{D}_{\mathscr{M}}(A, w, C, \lambda)) \right).$$

Moreover, since each instruction in ϕ corresponds to a production in $\hat{P}$, imply what $\overline{D}_{\mathscr{M}}(A, w, C, \lambda) = \hat{d}_{\mathscr{G}}(A, w)$, replacing in the previous equation we have,

$$deg_{\mathscr{M}}(w) = \bigoplus \left(\otimes(\hat{S}(A), \hat{d}_{\mathscr{G}}(A, w)) \right)$$

however, by definition $\mu_{\mathscr{G}}(w) = \bigoplus \left(\otimes(\hat{S}(A), \hat{d}_{\mathscr{G}}(A, w)) \right)$, thus $\text{deg}_{\mathscr{M}}(x) = \mu_{\mathscr{G}}(x)$, concluding that $L(\mathscr{M}) = L(\mathscr{G})$.

By symmetry of these two results and by Definition 18, we have the following corollary.

Corollary 1. *A fuzzy language $\mathbb{L}$ is a fuzzy linear language, if and only if there is a FLA $\mathscr{M}$ such that $L(\mathscr{M}) = \mathbb{L}$.*

Proof. Immediately by Definition 18 and the Theorems 3 and 4.

A FLA $\mathscr{M} = \langle Q_L, Q_R, \Sigma, \phi, \hat{I}, F, \otimes, \oplus \rangle$ will be said simplified, or abbreviated S-FLA, if $|F| = 1$, i.e., there is only one final state.

Theorem 5. *If $\mathscr{M}$ is a FLA, then there exist a S-FLA $\mathscr{M}'$ such that $L(\mathscr{M}') = L(\mathscr{M})$*

Proof. Suppose that $\mathscr{M} = \langle Q_L, Q_R, \Sigma, \phi, \hat{I}, F, \otimes, \oplus \rangle$ is a FLA, thus we build a S-FLA $\mathscr{M}' = \langle Q_L \cup \{q_f\}, Q_R, \Sigma, \phi', \hat{I}, \{q_f\}, \otimes, \oplus \rangle$ where for $a \in (\Sigma \cup \{\lambda\})$ we have ϕ' defined by:

$$\phi'(q,a,q') = \begin{cases} \phi(q,a,q'), & \text{if } q,q' \in (Q_L \cup Q_R) \\ 1, & \text{if } q \in F, q' = q_f \\ 0, & \text{else} \end{cases}$$

Now for all $(w,\sigma) \in L(\mathscr{M})$ we have $(q,w) \succ^*_{\mathscr{M}} (q',\lambda)$ with $q' \in F$ so by definition,

$$deg_{\mathscr{M}}(w) = \sigma \quad \Leftrightarrow \quad \bigoplus \Big(\otimes \big(\hat{I}(q), \overline{D}_{\mathscr{M}}(q,w,q',\lambda)\big)\Big) = \sigma.$$

But by construction of $\mathscr{M}'$, all the instructions in ϕ are also instructions in ϕ', moreover, is easy to see that in $\mathscr{M}'$ we have that $(q,w) \succ^*_{\mathscr{M}'} (q',\lambda) \succ_{\mathscr{M}'} (q_f,\lambda)$ and as by construction of $\mathscr{M}'$ for all $q \in F$ that $(q,q_f)_\lambda = 1$, so is easily see that,

$$\begin{aligned} \overline{D}_{\mathscr{M}'}(q,w,q_f,\lambda) &= \bigoplus \Big(\otimes \big(\overline{D}_{\mathscr{M}}(q,w,q',\lambda), 1\big)\Big) \\ &= \bigoplus \Big(\overline{D}_{\mathscr{M}}(q,w,q',\lambda)\Big) \\ &= \overline{D}_{\mathscr{M}}(q,w,q',\lambda). \end{aligned}$$

And thus,

$$\begin{aligned} deg_{\mathscr{M}'}(w) &= \bigoplus \Big(\otimes \big(\hat{I}(q), \overline{D}_{\mathscr{M}'}(q,w,q_f,\lambda)\big)\Big) \\ &= \bigoplus \Big(\otimes \big(\hat{I}(q), \overline{D}_{\mathscr{M}}(q,w,q',\lambda)\big)\Big) \\ &= deg_{\mathscr{M}}(w). \end{aligned}$$

Therefore, $deg_{\mathscr{M}'}(w) = deg_{\mathscr{M}}(w)$, so $L(\mathscr{M}') = L(\mathscr{M})$.

Remark 5. *For all S-FLA $\mathscr{M} = \langle Q_L, Q_R, \Sigma, \phi, \hat{I}, F, \otimes, \oplus \rangle$ by $|F| = \{q_f\}$ we can write,*

$$deg_{\mathscr{M}}(w) = \bigoplus_{q \in (Q_L \cup Q_R)} \big(\otimes(\hat{I}(q), \overline{D}_{\mathscr{M}}(q,w,q_f,\lambda))\big)$$

4.2 Fuzzy Nondeterministic 2-tapes Automata

The model of fuzzy nondeterministic 2-tapes automata is a machine with 2 memory tapes (read only). Each tape has a head of read initially located in the leftmost cell of the memory. Formally we define such automata as follows. This model is an extension of the model proposed by Rosenberg [17].

Definition 27. *Fuzzy Nondeterministic 2-tapes Automata (or FNTA), are structures of form $m = \langle Q, \Sigma, O, T, \upsilon, q_{acc}, \otimes, \oplus \rangle$ where,*

Q is a set finite of states;

Σ is a alphabet;

$O : Q \to [0,1]$ is a fuzzy set of origin states;

$T : Q \to \{1,2\}$ is a choose function of tape;

$\upsilon : Q \times (\Sigma \cup \{\lambda\}) \times Q \to]0,1]$ is a positive fuzzy semi-set of the instructions;

$q_{acc} \in Q$ is the acceptance state and

$\otimes$ is a t-norm and $\oplus$ is a t-conorm.

Let's write $\upsilon(q,a,q') = \sigma$ as $(q,q')^a = \sigma$ for all $a \in (\Sigma \cup \{\lambda\})$ and $q,q' \in Q$, and in this case we denote $(q,q')^a \in \upsilon$.

Example 6. *The structure $m = \langle\{q_0,q_1,q_2,q_{acc}\},\{a,b\},\{(q_0,1)\},T,\upsilon,q_{acc}.min,max\rangle$ with $T(q_0) = 1, T(q_1) = 2, T(q_2) = 1, T(q_{acc}) = 1$ and υ defined by $\psi(q_0,a,q_1) = 0.8, \psi(q_1,b,q_2) = 1, \psi(q_2,c,q_{acc}) = 0.7$ is an FNTA.*

A configuration in FNTA m is a tuple (q,w_1,w_2) with $q \in Q$ and $w_1,w_2 \in \Sigma^*$, where q represents the current state, w_1 what remains to read from word on tape 1 and w_2 what remains to read from word on tape 2.

Definition 28. *Let m be a FNTA, for all id (q,w_1,w_2) we define the sets*

i. $\lambda_{des}^{(q',w_1,w_2)} = \{(q',w_1,w_2) \mid (q,q')^\lambda \in \upsilon\}$.

ii. $L_{des}^{(q',w_1,w_2)} = \{(q',w_1',w_2) \mid w_1' = M_l(w_1)$ *and* $(q,q')^{R_l(w_1)} \in \upsilon\}$.

iii. $R_{des}^{(q',w_1,w_2)} = \{(q',w_1,w_2') \mid w_2' = M_l(w_2)$ *and* $(q,q')^{R_l(w_2)} \in \upsilon\}$.

iv. $\Omega(q',w_1,w_2) = \lambda_{des}^{(q',w_1,w_2)} \cup L_{des}^{(q',w_1,w_2)} \cup R_{des}^{(q',w_1,w_2)}$.

The relation $(q,w_1,w_2) \vdash_m (q',w_1',w_2')$ denotes a computation from (q,w_1,w_2) onto (q',w_1',w_2'). The membership value of computation from (q,w_1,w_2) onto (q',w_1',w_2') is given as follows.

Definition 29. *Given a FNTA m the membership value of a computation step $(q,w_1,w_2) \vdash_m (q',w_1',w_2')$ is obtained by function $f : Q \times \Sigma^* \times \Sigma^* \times Q \times \Sigma^* \times \Sigma^* \to [0,1]$ where f is:*

$$f(q,w_1,w_2,q',w_1',w_2') = \begin{cases} (q,q')^\lambda, & \textit{if } (q',w_1',w_2') \in \lambda_{des}^{(q,w_1,w_2)} \\ (q,q')^{R_l(w_1)}, & \textit{if } T(q) = 1 \textit{ and } (q',w_1',w_2') \in L_{des}^{(q',w_1,w_2)} \\ (q,q')^{R_l(w_2)}, & \textit{if } T(q) = 2 \textit{ and } (q',w_1',w_2') \in R_{des}^{(q',w_1,w_2)} \\ 0, & \textit{any other case} \end{cases} \tag{4.3}$$

The relation $\vdash_m^*$ is the reflexive and transitive closure of $\vdash_m$ and we extend the f function by the function $\hat{f}$ below.

Definition 30. *Let a FNTA m the value of membership value of relation* $(q,w_1,w_2) \vdash_m^* (q',w_1',w_2')$ *is obtained by function* $\hat{f} : Q \times \Sigma^* \times \Sigma^* \times Q \times \Sigma^* \times \Sigma^* \to [0,1]$ *where* $\hat{f}$ *is:*

$$\hat{f}(q,x,y,q',x',y') =$$

$$\begin{cases} f(q,x,y,q',x',y'), & \text{if } (q,w_1',w_2') \in \Omega^{(q,w_1,w_2)} \\ \bigoplus_{(q_i,x_1,y_1)\in\Omega^{(q,w_1,w_2)}} \left(\otimes(f(q,x,y,q_i,x_1,y_2),\hat{f}(q_i,x_1,y_2,q',x',y'))\right), & \text{if } \Omega^{(q,w_1,w_2)} \neq \emptyset \\ 0, & \text{any other case} \end{cases}$$

Example 7. *Given the FNTA of Example 6 the membership value of computation* $(q_0,ac,b) \vdash_m^* (q_{acc},\lambda,\lambda)$ *is,*

$$\begin{aligned} \hat{f}(q_0,ac,b,q_{acc},\lambda,\lambda) &= max\left(min(f(q_0,ac,b,q_1,b,c),\hat{f}(q_1,c,b,q_{acc},\lambda,\lambda))\right) \\ &= max(min(0.8,max(\otimes(1,0.7)))) \\ &= max(min(0.8,0.7)) \\ &= 0.7 \end{aligned}$$

There exist many possibilities of dividing a word w, thus for all w we define the set $\nabla_w = \{(x,y) \in \Sigma^* \times \Sigma^* \mid w = xy\}$. We define the membership value of m accept w as follows.

$$Acc_m(w) = \bigoplus_{(x,y)\in\nabla_w, q\in Q} \left(\otimes(O(q),\hat{f}(q,x,y^R,q_{acc},\lambda,\lambda))\right) \tag{4.4}$$

Definition 31. *The language of any FNTA m is a fuzzy set* $L(m) = \{(w,Acc_m(w)) \mid w \in \Sigma^*\}$.

We can then show a relation between S-FLA class machines and class FNTA machines.

Theorem 6. *Let* $m = \langle Q,\Sigma,O,T,\upsilon,q_{acc},\otimes,\oplus\rangle$ *be a FNTA. Then there exist an S-FLA* $\mathscr{M} = \langle Q_L,Q_R,\Sigma,\phi,O,\{q_{acc}\},\otimes,\oplus\rangle$ *such that* $L(\mathscr{M}) = L(m)$.

Proof. Let a FNTA $m = \langle Q,\Sigma,O,T,\upsilon,q_{acc},\otimes,\oplus\rangle$, we build a machine S-FLA $\mathscr{M} = \langle Q_L,Q_R,\Sigma,\phi,O,\{q_{acc}\},\otimes,\oplus\rangle$ where,

(i) $Q_L = \{q \in Q \mid T(q) = 1\} \cup \{q_{acc}\}$ and $Q_R = \{q \in Q \mid T(q) = 2\}$.

(ii) For all $q,q' \in Q$ and $a \in (\Sigma \cup \{\lambda\})$, $(q,q')_a = (q,q')^a$.

Clearly $\mathscr{M}$ is a S-FLA. Now for all $(w,\sigma) \in L(m)$ we have that,

$$Acc_m(w) = \sigma \quad \Leftrightarrow \quad \bigoplus_{(x,y)\in\nabla_w,q\in Q} \left(\otimes(O(q),\hat{f}(q,x,y^R,q_{acc},\lambda,\lambda))\right) = \sigma$$

Now notice that, by construction of the $\mathscr{M}$ for each instruction in m of the form $(q,q')^a$ there exist a instruction equivalent $(q,q')_a$ in $\mathscr{M}$. And for all $(x,y) \in \nabla_w$ and $q' \in Q$, since that $w = xy$ the first configuration (q',x,y^R) of the m correspond the first ID (q',xy) in $\mathscr{M}$, as every instruction in m has a instruction equivalent in $\mathscr{M}$ we have that for all computation $(q',x,y^R) \vdash_m^* (q_{acc},\lambda,\lambda)$ there exist a $(q',xy) \succ_{\mathscr{M}}^* (q_{acc},\lambda)$ equivalent in $\mathscr{M}$, i.e., $\overline{D}_{\mathscr{M}}(q',xy,q_{acc},\lambda) = \hat{f}(q',x,y^R,q_{acc},\lambda,\lambda)$, so,

$$\begin{aligned} deg_{\mathscr{M}}(w) &= \bigoplus_{(x,y)\in\nabla_w, q\in Q} \Big(\otimes\left(O(q),\overline{D}_{\mathscr{M}}(q,xy,q_{acc},\lambda)\right)\Big) \\ &= \bigoplus_{(x,y)\in\nabla_w, q\in Q} \Big(\otimes\left(O(q),\hat{f}(q',x,y^R,q_{acc},\lambda,\lambda)\right)\Big) \\ &= Acc_m(w). \end{aligned}$$

Therefore, $L(\mathscr{M}) = L(m)$.

Theorem 7. *Let $\mathscr{M} = \langle Q_L, Q_R, \Sigma, \phi, \hat{I}, F, \otimes, \oplus\rangle$ be a S-FLA. Then there is a FNTA $m = \langle Q, \Sigma, O, T, \upsilon, q_{acc}, \otimes, \oplus\rangle$ such that $L(m) = L(\mathscr{M})$.*

Proof. Suppose that $\mathscr{M} = \langle Q_L, Q_R, \Sigma, \phi, \hat{I}, F, \otimes, \oplus\rangle$ is a S-FLA, so $F = \{q_f\}$ with $q_f \in (Q_L \cup Q_R)$, thus we can build a FNTA $m = \langle Q_L \cup Q_R, \Sigma, \hat{I}, T, \upsilon, q_f, \otimes, \oplus\rangle$ where,

(i) For all $q \in (Q_L \cup Q_R)$ define $T(q) = \begin{cases} 1, & \text{if } q \in Q_L \\ 2, & \text{else} \end{cases}$

(ii) Define $\upsilon(q,a,q') = \phi(q,a,q')$, that is, $(q,q')^a = (q,q')_a$.

Clearly m is a FNTA. Now for all $(w,\sigma) \in L(\mathscr{M})$ by definition we have that,

$$deg_{\mathscr{M}}(w) = \sigma \Leftrightarrow \bigoplus_{q\in(Q_L\cup Q_R)} \left(\otimes(\hat{I}(q),\overline{D}_{\mathscr{M}}(q,w,q_f,\lambda))\right) = \sigma$$

Notice that, by construction of the m we have that for each instruction in $\mathscr{M}$ of the form $(q,q')_a$ there exist an instruction equivalent $(q,q')^a$ in m. Moreover, for all w there exist a set ∇_w, and as every instruction in $\mathscr{M}$ has an instruction equivalent in m we have that for all computation $(q',xy) \succ_{\mathscr{M}}^* (q_{acc},\lambda)$ there exist $(q',x,y^R) \vdash_m^* (q_{acc},\lambda,\lambda)$ equivalent in m, i.e., $\overline{D}_{\mathscr{M}}(q',xy,q_{acc},\lambda) = \hat{f}(q',x,y^R,q_{acc},\lambda,\lambda)$, and since $w = xy$ we have

$$\begin{aligned} Acc_m(w) &= \bigoplus_{(x,y)\in\nabla_w, q\in Q} \Big(\otimes\left(O(q),\hat{f}(q',x,y^R,q_{acc},\lambda,\lambda)\right)\Big) \\ &= \bigoplus_{(x,y)\in\nabla_w, q\in Q} \Big(\otimes\left(O(q),\overline{D}_{\mathscr{M}}(q,xy,q_{acc},\lambda)\right)\Big) \\ &= deg_{\mathscr{M}}(w). \end{aligned}$$

Therefore, $L(m) = L(\mathscr{M})$.

Corollary 2. *A fuzzy language $\mathbb{L}$ is a fuzzy linear language, if and only if there is a FNTA $m = \langle Q, \Sigma, O, T, \upsilon, q_{acc}\rangle$ such that $L(m) = \mathbb{L}$.*

Proof. Immediately by Corollary 1 and the Theorems 5, 6 and 7.

5 CONCLUSION

In this paper, we introduced the linear normal form (LNF) and the linear normal strong form (LNSF) for the class of linear fuzzy grammars. We show that these form preserves the degrees membership of generated words. We also propose two new fuzzy machines, which are equivalent in terms of language. Finally by the Corollaries 1 and 2 we can use FLA's and FNTA's as models to characterize the class of the linear fuzzy languages.

ACKNOWLEDGEMENT

Our most sincere thanks to the anonymous reviewers who with their comments only saw to enrich this work, and also, the first author thanks FAPERN and CAPES for the financial support.

RESUMO. Neste artigo, apresentamos duas novas formas normais para as gramáticas lineares fuzzy. Além disso, introduzimos duas novas classes de máquinas fuzzy. A primeira é a classe dos autômatos lineares fuzzy (FLA). Mostraramos que este tipo de autômato é equivalente as gramáticas lineares fuzzy, i.e, provamos que a classe das linguagens lineares fuzzy é reconhecida pelos autômatos lineares fuzzy. Por fim, introduzimos a classe dos autômatos não-determinísticos de 2-fitas fuzzy (FNTA) e mostramos a equivalência entre estes e os autômatos lineares fuzzy.

Palavras-chave: Gramáticas Fuzzy, Gramáticas Lineares Fuzzy, Linguagens Fuzzy, Autômatos Fuzzy.

REFERENCES

[1] C. Alsina, E. Trillas & L. Valverde. On non-distributive logical connectives for fuzzy set theory. *BUSEFAL*, **3** (1980), 18–29.

[2] V. Amar & G. Putzolu. On a Family of Linear Grammars. *Information and Control*, **7** (1964), 283–291.

[3] P.R.J. Asveld. Fuzzy context-free languages – Part 1: Generalized fuzzy context-free grammars. *Theoretical Computer Science*, **347** (2005), 167–190.

[4] B.R.C. Bedregal. λ-ALN: Autômatos Lineares Não-Determinísticos com λ-Transições. *TEMA – Tendências em Matemática Aplicada e Computacional*, **12**(3) (2011), 171–182.

[5] B.R.C. Bedregal. Nondeterministic Linear Automata and a Class of Deterministic Linear Languages. In Preliminary Proceedings LSFA (2015).

[6] B.R.C. Bedregal, B.M. Acióly & A. Lyra. *Introdução à Teoria da Computação: Linguagens Formais, Autômatos e Computabilidade*. EdUnP/FAPERN, Natal, RN, 1 edition (2010).

[7] B.R.C. Bedregal & S. Figueira. On the computing power of fuzzy Turing machines. *Fuzzy Sets and Systems*, **159**(9) (2008), 1072–1083.

[8] S.R. Chaudhari & D.D. Komejwar. On Fuzzy Regular Grammars. *Advances in Fuzzy Mathematics*, **6**(1) (2011), 89–104.

[9] N. Chomsky. Three models for the description of language. *IRE Transactions on Information Theory*, **2** (1956), 113–124.

[10] M.A. Harrison. *Introduction to Formal Languages Theory*. Addison-Wesley Publishing Company (1978).

[11] J.E. Hopcroft, R. Motwani & J.D. Ullman. *Introduction To Automata Theory, Languages, and Computation*. Addison-Wesley, New York, USA, 2 edition (2001).

[12] E.P. Klement, R. Mesiar & E. Pap. *Triangular Norms*, volume 8 of *Trends in Logic*. Springer-Science+Business Media, B.V., Poland (2000).

[13] E.T. Lee & L.A. Zadeh. Note on fuzzy languages. *Information Sciences*, **1** (1969), 421–434.

[14] A. Maciel. *Aplicações de Autômatos Finitos Nebulosos no Reconhecimento Aproximado de Cadeias*. Master's thesis, Escola Politécnica, Universidade de São Paulo, USP, São Paulo, SP (2006).

[15] K. Menger. Statistical metrics. *Proc. Nat. Acad.*, **28** (1942), 535–537.

[16] J.N. Mordeson & D.S. Malik. *Fuzzy Automata and Languages: Theory and Applications*. Chapman & Hall/CRC, Washington, D. C., USA (2002).

[17] A.L. Rosemberg. A machine realization of the linear context-free languages. *Information and Control*, **10** (1967), 175–188.

[18] B. Schweizer & A. Sklar. Associative functions and statistical triangle inequalities. *Publicationes Mathematicae Debrecen*, **8** (1961), 168–186.

[19] A.D. Silva Farias, V.S. Costa, R.H. Santiago & B.R.C. Bedregal. A Residuated Function in a Class of Mealy Type L-Valued Finite Automaton. In Fuzzy Information Processing Society (NAFIPS), 2016 Annual Conference of the North American (2016).

[20] A.D. Silva Farias, L.R. de Araújo Lopes, B.R.C. Bedregal & R.H. Santiago. Closure properties for fuzzy recursively enumerable languages and fuzzy recursive languages. *Journal of Intelligent & Fuzzy Systems*, **31**(3) (2016), 1795–1806.

[21] H. Xing, Q. Daowen & F. Liu. Automata theory based on complete residuated lattice-valued logic: Pushdown automata. *Fuzzy Sets and Systems*, **160**(8) (2009), 1125–1140.

[22] L.A. Zadeh. Fuzzy Sets. *Information and Control*, **8** (1965), 338–353.