

# A Novel Approach for the Determination of Membership Values of the Strings in Fuzzy Languages

Rahul Kumar Singh  
Dept. of CSE  
Chandigarh University  
Mohali, India

Ajay Kumar, Ph.D  
Dept. of CSE  
Thapar University  
Patiala, India

## ABSTRACT

Classical automata theory can not deal with uncertainty. To deal with uncertainty in system modeling, fuzzy automata come into the existence. Fuzzy automaton depends on membership value. For finding the membership degree of strings in fuzzy automata, the concept of max – min automaton and min – max automaton can be used. Many researchers have used the concept of max – min automaton for finding the fuzzy languages. In this paper, the concept of min – max automaton for finding the membership degree of strings in fuzzy automata has been used.

## General Terms

Theoretical computer science

## Keywords

Nondeterministic fuzzy automata, Fuzzy automata, Min – max automata

## 1. INTRODUCTION

Automata theory is firmly similar to formal language theory. The best known computational devices are Turing machine, finite automata, linear bounded automata, and pushdown automata. For modeling of formal systems computational devices are used. Computational devices are used to solve many problems in computer science such as description of natural languages, token finding, lexical analysis, digital system designing, string matching, and speech recognition etc.

Finite automata can not deal with system uncertainty. To deal with system uncertainty, fuzzy automata are used instead of finite automata. Fuzzy automata and language was first introduced by Santos in 1960 [6] and in 1967 Wee introduced mathematical formulation of fuzzy automata [20]. Fuzzy automata can move from one state to another (unlike classical case) with certain membership values. In 1969 [21] proposed Fuzzy language over an input alphabet  $\Sigma$ . Fuzzy automata have been widely used in various applications such as neural network, learning system, description of natural language, control system and many other areas [1]. Fuzzy automata can be described as deterministic, nondeterministic fuzzy automata. Many researchers describe fuzzy automata as deterministic fuzzy automata. Fuzzy automata can be categorized into three types based on fuzzy transition function : In the first type [24], fuzzy transition relations can be defined as  $\delta : Q \times \Sigma \rightarrow f(Q)$ , where  $Q$  represent the set of states, and  $f(Q)$  represents the set of all fuzzy subset of  $Q$ , and  $\Sigma$  represent the set of input alphabets. Fuzzy transition relation  $\delta : Q \times \Sigma \rightarrow f(Q)$  is equivalent to  $\delta' : Q \times \Sigma \times Q \rightarrow L$ , where  $L$  represent the lattice and  $L$  can be replaced

with interval  $[0,1]$  or other algebraic structure such as complete lattice, and lattice ordered monoid etc. According to the second type of fuzzy transition relation, the fuzzy transition function of fuzzy automata is similar to the transition function of deterministic finite automata and captures the uncertainty by fuzzy final and initial states. In the third type, new components can be added in the first type. Fuzzy automata are similar to mealy finite machine. Fuzzy automata fully dependent on membership value and these values can be taken with complete lattice, lattice ordered monoid, and many other algebraic structure.

In the fuzzy automata, compositional inference methods are used and these methods classify fuzzy finite automata into three types such as max – product automata, min – max automata, max – min automata [3]. Many researchers have used the concept of max – min automata to find the fuzzy language [3], [24]. In this paper, the concept of min – max automata to find the fuzzy language has been used.

## 2. RELATED WORK

In this section, a systematic review is accomplished on many researches in the field of Fuzzy Automata based on many criterions such as algebraic aspects, minimization of fuzzy automata, closure properties of fuzzy automata, application of fuzzy automata, and conversion of fuzzy regular expressions to fuzzy automata.

### 2.1 Algebraic Aspects

Li and Pedrycz [25] used the concept of lattice – ordered monoids for constructing the fuzzy automata with membership value. Ignjatovic *et al.* [11] explained determination of fuzzy finite automata with membership degree using complete residuated lattice.

Tiwari and Sharan [18] defined the concept of fuzzy finite automata with membership degree using lattice – ordered monoids. For fuzzy finite automata, they explained some topological and algebraic concepts. Complete lattice, lattice – ordered monoid, complete residuated lattice, and other algebraic structures can be used in fuzzy automata. Various algebraic aspects can be used for finding these membership values.

### 2.2 Minimization of Fuzzy Automata

Malik *et al.* [4] proposed the approach for minimization of fuzzy finite automata using the homomorphism image property. Topencharov and Peeva [19] proposed a new algorithm for minimization and reduction of fuzzy finite automata. Ignjatovic *et al.* [11] proposed an efficient approach for the construction of smaller fuzzy finite automaton using the fuzzy right congruence. Ciric *et al.* [15] used bi-simulation for reducing the number of states in fuzzy automata. They explained the uniform fuzzy relation, state reduction,

equivalence of automata, backward and forward simulation. Ciric *et al.* [14] explained the alternating reduction by using two approach *i.e* left invariant fuzzy finite equivalence and right invariant fuzzy finite equivalence. In this paper they explained efficient steps for finding the fuzzy equivalence. They explained complete residuated lattice, factor fuzzy automaton, alternate reduction, fuzzy relation equation, and state reduction. Cheng and Mo [22] explained minimization algorithm, mealy type of fuzzy automata, minimization, and fuzzy automata. Using equivalence relations, they reduce the number of states. Many such approaches are proposed in the literature for the minimization of fuzzy automata.

### 2.3 Closure Properties

Mizumoto *et al.*[12] explained closure properties of fuzzy automata. Li and Pedrycz [25] proposed the concept of closure properties for deterministic and nondeterministic fuzzy automata. They proved that nondeterministic fuzzy automata are not closed under the Complement operation and the deterministic fuzzy automata are not closed under the Kleene closure operation. They proved that the power of nondeterministic fuzzy automata is more than the deterministic fuzzy automata. Li *et al.* [26] explained relationship between several types of fuzzy finite automata. In relationship they include nondeterministic lattice ordered monoid fuzzy automata with  $\mathcal{E}$ - move, nondeterministic lattice ordered monoid fuzzy automata without  $\mathcal{E}$ - move, deterministic fuzzy automata using lattice ordered monoid, and lattice ordered monoid fuzzy finite state machine. They also compared the power of all the fuzzy automata Application of fuzzy automata. Bedregal and Figueira [35] explained that the power fuzzy Turing machine is more than the classical Turing machine and fuzzy Turing machine are more efficient also. In this paper, they explained universal machine, recursively enumerable set, and fuzzy function.

### 2.4 Application of Fuzzy Automata

Wee and Fu [21] introduced the model of learning system using the fuzzy automata. Rigatos [8] suggested that a fuzzy automaton can be used in the system modelling. He proposed two approaches named as the syntactic analysis and fuzzy automata for the fault detection. Blanco *et al.* [2] suggested that how we can infer fuzzy regular grammar using two layers neural – network architecture from fuzzy examples. They explained the fuzzy regular grammar, fuzzy recurrent neural network, and fuzzy grammatical inference. Fuzzy automata can be used in the artificial intelligence. Astrain *et al.* [10] suggested how to measure the fuzziness between the strings. Wu *et al.* [16] explained the fuzzy automata system to achieve improved image processing and target recognition. Using fuzzy finite automata system, firstly good image processing achieved and then carried out for the target recognition. The fuzzy finite automata system comprises four factors *i.e* experiment, factor extraction, target machine and image processing. In this paper, they used two features *i.e* local and global of target image and using fuzzy automata system, target recognition accomplished. They explained image processing, target recognition, fuzzy automata and proved that results of recognizing the target is more than 94.59%. Bailador and Trivino [28] introduced temporal fuzzy automata for pattern recognition. They explained pattern recognition, fuzzy models, and hidden Markov model. Pattern recognition using temporal fuzzy automata is more efficient than pattern recognition using hidden Markov model. Ciric *et al.*[27] explained four types of bi-simulation *i.e* (backward, forward, backward – forward and forward – backward bi-simulation)

and two types of simulation *i.e* (backward, forward) in fuzzy finite automata. They explained complete residuated lattice, fuzzy relation, post – fixed point, fuzzy relation inequality, bi-simulation, simulation, and fuzzy automata. In this paper, they explained an efficient approach for finding whether there is a bi-simulation/simulation between fuzzy automata and if there exists bi-simulation/ simulation between fuzzy automata then find out the greatest bi-simulation/simulation. Many researchers are still exploring the various properties of fuzzy automata.

### 2.5 Conversion of Fuzzy Regular Expression to Fuzzy Automata

Stamenkovic and Ciric [1] introduced the new approach for conversion of fuzzy regular expressions to fuzzy finite automata. In this paper, they used the concept of position automata for the conversion of fuzzy regular expressions to fuzzy finite automata. Li and Pedrycz [25] suggested the conversion of fuzzy regular expressions to fuzzy automata, but there is a scope of improvement in the efficiency of the conversion.

### 2.6 Various Terminologies and Researches related to the Fuzzy Automata

Santos [6] suggested the concept of max-min automata. In this paper, they explained the different classes of automata, behaviour of max-min automata, homomorphism, and equivalence of max-min automata. Ignjatovic *et al.* [29] introduced myhill – Nerode theorem for the fuzzy automata and language. They explained the relationship between fuzzy language, deterministic automata and fuzzy automata.

Xing [30] described the concepts of fuzzy context – free languages, fuzzy pushdown automata, and fuzzy context – free grammar. In this paper, they explained the behaviour and movement of multi-stack and one – stack fuzzy pushdown automata (FPDA) for recognizing the fuzzy languages. Tiwari and Srivastava[31] suggested the decomposition of fuzzy automata in a unique form. In this paper, they explained the concept of source – splitting sub-automaton and decomposable sub-automaton.

Doostfateme and Kremer [32] described that classical automata theory deals with discrete space, whereas fuzzy automata theory deals with continuous space. Classical automata cannot manage the fuzziness in system modelling. In order to manage fuzziness in any system, fuzzy logic is incorporated with the automata theory, and the resulted fuzzy automata can deal with fuzziness in any of the system. In this paper, they explained that the deterministic finite automata (DFA) are widely used in many applications, but they have some issues such as they do not deal with fuzziness etc. In particular, they emphasized on the multi membership resolution.

Li [36] explained the fuzzy Turing machine using distributive lattice with membership value that is also called lattice ordered fuzzy Turing machine. In this paper, he explained deterministic fuzzy Turing machine, nondeterministic turing machine, recursively enumerable language, universal machine and fuzzy system model. He also introduced the approach of lattice – ordered nondeterministic fuzzy polynomial time computations, lattice – ordered deterministic fuzzy polynomial time computations.

Benlahcen and Lamotte [33] defined fuzzy automata and recalled some functional properties and theorem of fuzzy logic that are used in fuzzy automata. They proposed the synthesis method of fuzzy automata and used numerical for the illustration of the efficiency of synthesis method.

Santos [34] proposed the concept of fuzzy automata and language. In this paper, they explained how fuzzy automata theory is similar to classical automata theory. They used max – product composition in fuzzy automata for generating the fuzzy language.

### 3. PRILIMINARIES

In this section, the concepts of fuzzy subsets, fuzzy language, and fuzzy automata are explained.

The concept of fuzzy subsets of a set was given in 1965 by Zadeh [7]. Let A be a set and B is subset of A, denoted by  $B \subseteq A$ . The characteristics function of B is a function  $A_B$  of A into  $\{0, 1\}$  defined below,

$$A_B(x) = \begin{cases} 1 & \text{if } x \in A, \\ 0 & \text{otherwise,} \end{cases}$$

#### 3.1 Fuzzy Subset

Please use a 9-point Times Roman font, or other Roman font with serifs, as close as possible in appearance to Times Roman in which these guidelines have been set. The goal is to have a 9-point text, as you see here. Please use sans-serif or non-proportional fonts only for special purposes, such as

A fuzzy subset A of X is a function of X into the closed interval  $[0, 1]$  [7], [9].

Let A be the fuzzy subset of set X and  $\forall x \in X$ , and where  $A(x)$  have some membership degree.

The support of a fuzzy subset A is defined in following way [24]:

$$\text{Supp}(A) = \{ x \in X \mid A(x) > 0 \}.$$

If membership degree is zero on X, that means support is empty and fuzzy set is also empty. According to Zadeh, if support (A) is finite then A can be written in following way [23]:

$$A = \frac{A(x_1)}{x_1} + \frac{A(x_2)}{x_2} + \dots + \frac{A(x_n)}{x_n}.$$

This is a Zadeh's notation, where  $A(x_1)$  represents the membership degree at  $x_1$ . If it has a singleton in X that means  $\frac{1}{x}$ , where 1 is membership value at x and membership value zero in all the other elements of X. To find the height of A, where  $\forall A \in f(x)$  ( $f(x)$  represent the set of all fuzzy subsets of X) is defined in following way:

$$\text{height}(A) = \bigvee_{x \in X} A(x).$$

Suppose P and Q are representing two sets, given  $P, Q \in f(x)$  and  $\forall x \in X$ .

The intersection of P and Q is defined in following way

$P \cap Q$ , and membership function is defined in following way:

$$(P \cap Q) = P(x) \wedge Q(x).$$

The union of P and Q is defined as  $P \cup Q$ , and membership function is defined in following way:

$$(P \cup Q) = P(x) \vee Q(x).$$

The scalar multiplication  $\lambda \cdot P$  (where  $\lambda \in [0, 1]$ ) of  $\lambda$  and P defined as:

$$(\lambda \cdot P)(x) = \lambda \wedge P(x).$$

#### 3.2 Fuzzy Automata

Fuzzy automaton [3], [17] M has five tuple  $M = (Q, \Sigma, \delta, q_0, F)$ , Where

- Q is a finite non – empty set of states.
- $\Sigma$  is finite non – empty set of input alphabets.
- $\delta : Q \times \Sigma \times Q \rightarrow [0, 1]$  is a fuzzy transition function.
- $q_0 \in Q$ , where  $q_0$  is fuzzy initial state.
- $F \subseteq f(Q)$  is fuzzy subset of final states.

In fuzzy automata, fuzzy transition function  $\delta$  plays an important role. Fuzzy transition function  $\delta$  takes input alphabet  $\Sigma$  at present state and returns with membership value at Q. Suppose  $q_0, q_1 \in Q$  and  $a \in \Sigma$ , then fuzzy transition function define as  $\delta(q_0, a)(q_1)$ , a input is taken on state  $q_0$  and state  $q_1$  is reached with certain possibility. The fuzzy automaton M consider as *deterministic fuzzy automaton* because automaton in state  $q_0$  with input alphabet a gives rise to a unique possibility distribution on Q [13].

#### 3.3 Nondeterministic Fuzzy Automata

Nondeterministic fuzzy automata M has five tuple  $M = (Q, \Sigma, \delta, q_0, F)$ , where

- Q is a set finite non – empty set of states.
- $\Sigma$  is finite non – empty set of input alphabets.
- $\delta: Q \times (\Sigma \cup \{\mathcal{E}\}) \rightarrow 2^{f(Q)}$  is a fuzzy transition function.
- $q_0 \in Q$ , where  $q_0$  is fuzzy initial state.
- $F \subseteq f(Q)$  is fuzzy subset of final states.

All the tuple except fuzzy transition function are same as deterministic fuzzy automata. Fuzzy transition function of nondeterministic fuzzy automata is defined as: state  $q_0$  takes input a from  $\Sigma$  and chooses any one of the possibility distribution, and then reach at state  $q_1$  with certain membership degree. In nondeterministic fuzzy automaton  $\mathcal{E}$  - move allowed.

The special case of nondeterministic fuzzy automata is deterministic fuzzy automaton. The main difference between deterministic and nondeterministic fuzzy automaton is fuzzy transition function. In nondeterministic, more than one transition on the same input alphabet can be taken or choices are allowed in states: like  $\delta(q_0, a) = \{q_0, q_1, q_2 \dots q_n\}$ , a input is taken on state  $q_0$  and any of  $q_0, q_1, q_2 \dots q_n$  as the next state is chosen. In deterministic, choices are not allowed hence every state  $q_0$  in deterministic fuzzy automaton always has exactly one transition [23].

### 4. PROPOSED ARCHITECTURE

In this section, the concept of min – max automata to find the language of machine has been used. To explain the definition of nondeterministic fuzzy automata, let us consider the following example where fuzzy transition function and fuzzy language has been explained.

Example 1: Consider the fig.1, nondeterministic fuzzy automata where  $M = (Q, \Sigma, \delta, q_0, F)$ , where input alphabet  $\Sigma$

$= \{a, b\}$ , initial state is  $q_0$ , and final state is defined as  $F = \frac{0.4}{q_2} + \frac{0.8}{q_4}$ .

The extended fuzzy transition function for nondeterministic fuzzy automaton is defined in following way as:

Fuzzy transition function of nondeterministic fuzzy automata  $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow 2^{f(Q)}$  can be written in this way  $Q \times \Sigma^* \rightarrow 2^{f(Q)}$ .

$$\delta(q_0, \varepsilon) = \left\{ \frac{1}{q_0} \right\},$$

$$\delta^*(q_0, sb) = \delta(\delta^*(q_0, s), b) = \{ \delta(q_0, s)(q_1) \cdot \delta(q_1, b) \}, \text{ where } q_0, q_1 \in Q, \forall s \in \Sigma^* \text{ and } \forall b \in \Sigma.$$

The extended fuzzy transition function for nondeterministic fuzzy automaton is defined in following way as:

$$\delta(q_0, \varepsilon) = \left\{ \frac{1}{q_0} \right\},$$

$$\delta^*(q_0, sb) = \delta(\delta^*(q_0, s), b) = \bigcup_{q_1 \in Q} \{ \delta(q_0, s)(q_1) \cdot \delta(q_1, b) \}$$

where  $q_0, q_1 \in Q, \forall s \in \Sigma^*$  and  $\forall b \in \Sigma$ , and  $1/q_0$  is a singleton in  $Q$ .

To find the fuzzy languages  $L(M)$  of machine, on basis of min – max automata is defined as:

Given  $\forall s \in \Sigma^*$ , where  $s$  is a string,  $F$  is a set of final states,  $\wedge$  and  $\vee$  denote infimum and supremum respectively.  
 $L(M)(s) = \wedge_{q \in Q} \{ \text{height}(\delta^*(q_0, s) \vee F) \}$ .

For fig.1, fuzzy transition function is defined in the following way:

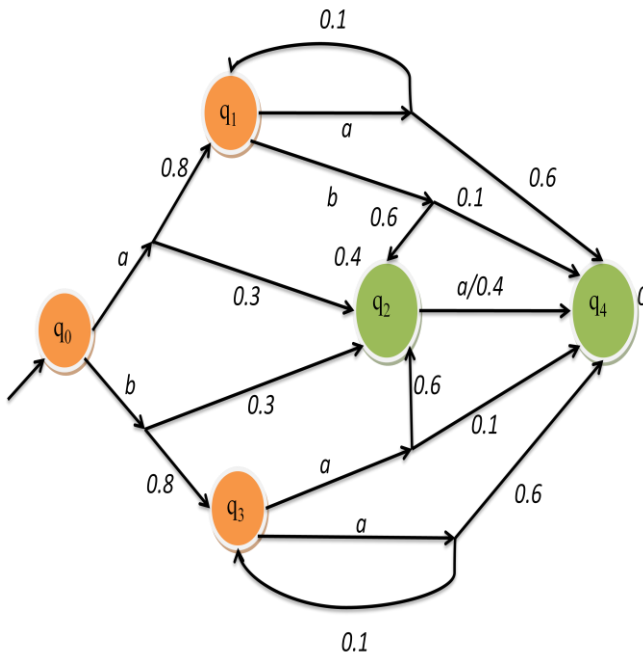


Fig.1: Nondeterministic fuzzy automaton.

$$\delta(q_0, \varepsilon) = \left\{ \frac{1}{q_0} \right\},$$

$$\delta(q_0, a) = \{ \delta(q_0, \varepsilon) \cdot \delta(q_0, a) \} = \left\{ \frac{0.8}{q_1} + \frac{0.3}{q_2} \right\},$$

$$\delta(q_0, b) = \{ \delta(q_0, \varepsilon) \cdot \delta(q_0, b) \} = \left\{ \frac{0.8}{q_3} + \frac{0.3}{q_2} \right\},$$

$$\delta^*(q_0, aa) = \{ \delta^*(\delta(q_0, a), a) \} = \left\{ \frac{0.1}{q_1} + \frac{0.6}{q_4}, \frac{0.3}{q_2} \right\},$$

$$\delta^*(q_0, ab) = \{ \delta^*(\delta(q_0, a), b) \} = \left\{ \frac{0.1}{q_4} + \frac{0.1}{q_2} \right\}.$$

Table 1. Fuzzy transition function for nondeterministic fuzzy automata.

$\delta$	$a$	$b$
$q_0$	$\left\{ \frac{0.8}{q_1} + \frac{0.3}{q_2} \right\}$	$\left\{ \frac{0.8}{q_3} + \frac{0.3}{q_2} \right\}$
$q_1$	$\left\{ \frac{0.1}{q_1} + \frac{0.6}{q_4} \right\}$	$\left\{ \frac{0.6}{q_2} + \frac{0.1}{q_4} \right\}$
$q_2$	$\left\{ \frac{0.4}{q_4} \right\}$	$\phi$
$q_3$	$\left\{ \frac{0.6}{q_2} + \frac{0.1}{q_4}, \frac{0.1}{q_3} + \frac{0.6}{q_4} \right\}$	$\phi$
$q_4$	$\phi$	$\phi$

Using min – max automaton for Fuzzy language string is accepted with some membership value.

$$L(M)(s) = \wedge_{q \in Q} \{ \text{height}(\delta^*(q_0, s) \vee F) \}.$$

$$L(M)(\varepsilon) = \wedge_{q \in Q} \{ \text{height}(\delta^*(q_0, \varepsilon) \vee F) \} = 0,$$

$$L(M)(a) = \wedge_{q \in Q} \{ \text{height}(\delta^*(q_0, a) \vee F) \} = 0.8,$$

$$L(M)(b) = \wedge_{q \in Q} \{ \text{height}(\delta^*(q_0, b) \vee F) \} = 0.8,$$

$$L(M)(aa) = \wedge_{q \in Q} \{ \text{height}(\delta^*(q_0, aa) \vee F) \} = 0.4,$$

$$L(M)(ab) = \wedge_{q \in Q} \{ \text{height}(\delta^*(q_0, ab) \vee F) \} = 0.8.$$

$L(M)(s)$  is the membership value in which the string  $s$  is accepted in fuzzy automata.

## 5. CONCLUSION AND FUTURE WORK

In this paper, the concept of min – max automaton for finding the membership value of strings in fuzzy automata has been used. In classical automata, strings are either accepted or rejected but in fuzzy automata, strings are accepted with certain membership values. In Future, work can be done in following directions:

- The max product can be used for finding fuzzy language.
- Design of an efficient approach for the conversion of Fuzzy regular expressions to Fuzzy automata.
- Pumping lemma for fuzzy language.
- Construct the fuzzy regular expression from fuzzy automata.

## 6. REFERENCES

[1] Stamenkovic, A. and Ciric, M. (2012) “Construction of fuzzy automata from fuzzy regular Expressions”, *Fuzzy Sets and Systems*, vol. 199, pp. 1-27.

[2] Blanco, Delgado, M. and Pegalajar (2001) “Fuzzy automaton induction using neural network”, *International Journal of Approximate Reasoning*, vol. 27, pp. 1 – 26.

- [3] Malik, D. S., Mordeson, J. N. and Sen, M. K. (1969) “On fuzzy regular language”, *Information Science*, vol. 88, pp. 263 – 273.
- [4] Malik, D. S., Mordeson, J. N. and Sen, M. K. (1999) “Minimization of fuzzy finite automata”, *Information Science*, vol. 113, pp. 323 – 330
- [5] Qiu, D.W. (2004) “Characterizations of fuzzy finite automata”, *Fuzzy Sets System*, vol. 141 pp. 391–414.
- [6] Santos, E. S. (1968) “Maximin automata”, *Info. Control*, vol. 12, pp. 367-377.
- [7] Lee, E.T. and Zadeh, L.A. (1969) “Note on fuzzy languages”, *Information Sciences*, vol. 1, pp. 421- 434.
- [8] Rigatos, G. G. (2009) “Fault detection and isolation based on fuzzy automata”, *Information Science*, vol. 179, pp. 1893 – 1902.
- [9] Mordeson, J. N. and Malik, D.S. (2002) “Fuzzy Automata and Languages: Theory and Applications”, *Chapman & Hall, CRC, Boca Raton, London*.
- [10] Astrain, J. J., Mendivil, J. R. G. and Garitagoitia, J. R. (2006) “Fuzzy automata with  $\varepsilon$ -moves compute fuzzy measures between strings”, *Fuzzy Sets and Systems*, vol. 157, pp. 1550 – 1559.
- [11] Ignjatovic, J., Ciric, M., Bogdanovic, S. and Petkovic, T. (2010) “Myhill–Nerode type theory for fuzzy languages and automata”, *Fuzzy Sets and Systems*, vol. 161, pp. 1288 – 1324.
- [12] Mizumoto, M., Toyoda, J. and Tanaka, K. (1969) “Some Considerations on Fuzzy Automata”, *Journal of Computer and System Science*, vol. 3, pp. 409 – 422.
- [13] Doostfateme, M. and Kremer, S.C. (2005) “New directions in fuzzy automata”, *International Journal of Approximate Reasoning*, vol. 38, pp. 175 – 214.
- [14] Ciric, M., Stamenkovic, A., Ignjatovic, J. and Petkovic, T. (2010) “Fuzzy relation equation and reduction of fuzzy automata”, *Journal of Computer and System Sciences*, vol. 69, pp. 609 – 633.
- [15] Ciric, M., Ignjatovic, J., Damljanovic, N. and Basic, M. (2012) “Bisimulations for fuzzy automata”, *Fuzzy Sets and Systems*, vol. 86, pp. 100 – 139.
- [16] Wu, Q – E., Pang, X – M. and Han, Z – Y. (2011) “Fuzzy automata system with application to target recognition based on image processing”, *Computer and Mathematic with application*, vol. 61, pp. 1267 – 1277.
- [17] Belohlavel, R. (2002) “Determinism and fuzzy automata”, *Information Sciences*, vol. 143, pp. 205 – 209.
- [18] Tiwari, S. P. and Sharan, S. (2012) “Fuzzy Automata Based on Lattice-ordered Monoid with Algebraic and Topological Aspects”, *Fuzzy Information Eng.* Vol. 2, pp. 155 – 164.
- [19] Topencharov, V. V. and Peeva, K. G. (1981) “Equivalence, Reduction and Minimization of Finite Fuzzy-Automata”, *Journal of Mathematical Analysis and Applications*, vol. 84, pp. 270 – 281.
- [20] Wee, W. G. (1967) “On Generalizations of Adaptive Algorithm and Application of the Fuzzy Sets Concept to Pattern Classification”, Ph.D. Thesis, Purdue University.
- [21] Wee, W.G. and Fu, K.S. (1969) “A formulation of fuzzy automata and its application as a model of learning systems”, *IEEE Transactions Systems Man cybernetics*, vol. 5, pp. 215–223.
- [22] Cheng W. and Mo Z – W. (2004) “Minimization algorithm of fuzzy finite automata,” *Fuzzy Set and System*, vol. 141, pp. 439 – 448.
- [23] Li, Y. and Pedrycz, W. (2004) “Regular expressions with truth values in lattice-monoid and their languages”, *Proc NAFIPS ‘04*, vol. 02, pp. 572 – 577.
- [24] Cao, Y. and Ezawa, Y. (2012) “Nondeterministic fuzzy automata”, *Information Sciences*, vol. 191, pp. 86-97.
- [25] Li, Y. M. and Pedrycz, W. (2005) “Fuzzy finite automata and fuzzy regular expressions with membership values in lattice-ordered monoids”, *Fuzzy Sets System*, vol.156, pp. 68–92.
- [26] Li, Z., Li, P. and Li, Y. (2006) “The relationships among several type of fuzzy automata”, *Information Sciences*, vol. 176, pp. 2208 – 2226.
- [27] Ciric, M., Ignjatovic, J., Jancic, I. and Damljanovic, N. (2012) “Computation of the greatest simulations and bisimulations between fuzzy automata”, *Fuzzy Sets and Systems*, vol. 208, pp. 22 – 42.
- [28] Bailador, G. and Trivino, G. (2010) “Pattern Recognition Using Temporal Fuzzy Automata”, vol. 161, pp. 37 – 55.
- [29] Ignjatovic, J., Ciric, M., Bogdanovic, S. and Petkovic, T. (2010) “Myhill–Nerode type theory for fuzzy languages and automata”, *Fuzzy Sets and Systems*, vol. 161, pp. 1288 – 1324.
- [30] Xing, H. (2007) “Fuzzy pushdown automata,” *Fuzzy Sets and Systems*, vol. 158, pp. 1437 – 1449.
- [31] Tiwari, S. P. and Srivastava, A. K. (2005) “On a decomposition of fuzzy automata”, *Fuzzy Sets and Systems*, vol. 151, pp. 503 – 511.
- [32] Doostfateme, M. and Kremer, S.C. (2005) “New directions in fuzzy automata”, *International Journal of Approximate Reasoning*, vol. 38, pp. 175 – 214.
- [33] Benlahcen, D. and Lamotte, M. (1981) “A Fuzzy Automaton Synthesis Method”, *International Federation of Automatic Control*, vol. 17, pp. 299 – 306.
- [34] Santos, E. S. (1976) “Fuzzy automata and languages”, *Information Sciences*, vol. 10, pp. 193–197.
- [35] Bedregal, B. C. and Figueira, S. (2008) “On the Computing Power of fuzzy Turing machine”, *Fuzzy Set and System*, vol. 159, pp. 1072 – 1083.
- [36] Li, Y. (2009) “Lattice valued fuzzy Turing machine computing power universality and efficiently”, *Fuzzy Set and System*, vol. 160, pp. 3453 – 3474, 2009.