2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)
July 6-11, 2014, Beijing, China

Lattice-valued Fuzzy Residual Finite Automata

Fugang Zhang[1], Yongming Li[2]

[1]School of Mathematics and Statistics
Huangshan University, Huangshan, 245021, China
Email: fugang@hsu.edu.cn
[2]College of Mathematics and Information Science, College of Computer Science
Shaanxi Normal University, Xi'an, 710062, China
Email: liyongming@snnu.edu.cn

***Abstract*—In this paper, we introduce the notion of lattice-valued fuzzy residual finite automaton ($LRFA$) and the $LRFA$-regular language with membership values in a complete residuated lattice. Next, we define saturation operator and reduction operator on lattice-valued finite automata(LFA), which provide a way to simplify $LRFA$ based on their closure properties in $LRFA$. At last, we define the canonical $LRFA$ based on the notion of irreducible residual language, prove that every $LRFA$-regular language is recognized by a unique canonical $LRFA$ which has a minimal number of states and largest initial and transition functions.**

I. Introduction

Automata Minimization of automata is one of the most important research fields of automata theory. Deterministic and nondeterministic finite automata are computationally equivalent, and nondeterministic finite automata can offer exponential state savings compared to deterministic ones. Contrary to the problem of minimizing $DFAs$, which is efficiently possible, the minimization of NFA is computationally hard, namely NP-complete. Denis, Lemay and Terlutte defined an important type of finite automata, residual finite state automata ($RFSA$), based on Myhill- Nerode theorem in [1,2], proved that for every regular language L, there exists a unique minimal $RFSA$ that recognizes L and which has both a minimal number of states and a maximal number of transitions.

As for the classical automata, the minimization of fuzzy automata is still an important issue. Unlike deterministic fuzzy automata, whose minimization is efficient, the problem of minimization of non-determinatic fuzzy automata is still NP-complete. The minimization of fuzzy automata was researched by left and right invariant fuzzy equivalences with membership values in a complete residual lattice in [23], the minimization of fuzzy automata with membership values in a lattice-ordered monoid was researched by Li and Lei in [19,21], the minimization by considering bisimulations for fuzzy automata was researched in [27], and the quotient minimization of fuzzy automata was researched in [26]. The purpose of the present paper is to provide another effective method for the minimization of fuzzy finite automata.

We consider the minimization of fuzzy finite automata by defining fuzzy residual finite automata. First, we introduce the notion of lattice-valued fuzzy residual finite automata ($LRFA$): an $LRFA$ is a lattice-valued fuzzy finite automata (LFA) in which its states define lattice-valued residual languages of the lattice-valued language that it recognizes, and the notion of $LRFA$-regular language which is an lattice-valued language recognized by an $LRFA$ with membership values in a complete residuated lattice, and we discuss some properties of the $LRFA$. Second, we study saturation operator and reduction operator of lattice-valued fuzzy finite automata, and obtain some useful results. It shows that $LRFA$s are closed under saturation operator and reduction operator. Last, we define the canonical $LRFA$ based on the notion of irreducible lattice-valued residual languages, and prove that every $LRFA$-regular language is recognized by a unique canonical $LRFA$ which has a minimal number of states and largest initial and transition functions.

The organization of this paper is as follows. In section 2, we introduce the notion of lattice-valued fuzzy finite automata, L-language and other basic concepts. In section 3, we define the lattice-valued residual finite automata ($LRFA$) and the $LRFA$-regular language, discuss some properties of $LRFA$. In section 4, we define saturation operator and reduction operator on lattice-valued finite automata. In section 5, we define the canonical $LRFA$, prove that every $LRFA$-regular language is recognized by a unique canonical $LRFA$. In section 6, we summarize this paper.

II. Preliminaries

In this paper we will use complete residuated lattices as the structures of membership values. A residuated lattice is an algebra $\mathscr{L} = (L, \wedge, \vee, \otimes, \rightarrow, 0, 1)$ such that

(L1) $(L, \wedge, \vee, 0, 1)$ is a lattice with the least element 0 and the greatest element 1,
(L2) $(L, \otimes, 1)$ is a commutative monoid with the unit 1,
(L3) $\otimes$ and $\rightarrow$ form an adjoint pair, i.e., they satisfy the adjunction property: for any $x, y, z \in L$,

$$x \otimes y \leq z \Leftrightarrow x \leq y \rightarrow z.$$

If, in addition, $(L, \wedge, \vee, 0, 1)$ is a complete lattice, then $\mathscr{L}$ is called a complete residuated lattice. From now on we assume that $\mathscr{L}$ is a complete residuated lattice.

It can be easily verified that with respect to $\leq$, $\otimes$ is monotonic in both arguments, $\rightarrow$ is monotonic in the second and anti-monotonic in the first argument, and for any $x, y, z \in L$ and any $\{x_i\}_{i\in I}, \{y_i\}_{i\in I} \subseteq L$, the following hold:

(1) $\bigvee_{i\in I}(x \otimes x_i) = x \otimes \bigvee_{i\in I} x_i, \quad \bigvee_{i\in I}(x_i \otimes x) = \bigvee_{i\in I} x_i \otimes x,$

(2) $(\bigvee_{i\in I} x_i) \to x = \bigwedge_{i\in I}(x_i \to x),$

(3) $x \to (\bigwedge_{i\in I} x_i) = \bigwedge_{i\in I}(x \to x_i),$

(4) $(x \otimes y) \to z = x \to (y \to z),$

(5) $(x \to y) \otimes (x \to z) \leq x \to z.$

For other properties of complete residuated lattices we refer to [29,30].

The most studied and applied structures of truth values, defined on the real unit interval [0, 1] with $x \wedge y = min(x, y)$, $x \vee y = max(x, y)$, are equipped with $\otimes$ being a left-continuous t-norm and the residuum is defined by $x \to y = \bigvee\{u \in [0, 1] | u \otimes x \leq y\}$. For example, the well-known structures are the Lukasiewicz structure $(x \otimes y = max(x + y - 1, 0), x \to y = min(1 - x + y, 1))$, product structure ($x \otimes y = x \cdot y, x \to y = 1$ if $x \leq y$ and $= y/x$ otherwise) and Gödel structure ($x \otimes y = min(x, y)$, $x \to y = 1$ if $x \leq y$ and $= y$ otherwise).

Definition 2.1([16]): A lattice-valued fuzzy finite automaton (LFA) is a 5-tuple, $\mathscr{A} = (Q, \Sigma, \delta, I, F)$, where Q, Σ are two finite nonempty sets, $\delta : Q \times \Sigma \times Q \longrightarrow L$ is an L-fuzzy subset of $Q \times \Sigma \times Q$, and $I, F : Q \longrightarrow L$ are L-fuzzy sets of Q. The elements of Q are called states, and the elements of Σ are called input symbols, respectively. δ is called a fuzzy transition function, and I, F are called fuzzy initial and final function, respectively.

Let Σ^* denote the set of all words of finite letters over Σ. ε denotes the empty word. For $\theta \in \Sigma^*$, $|\theta|$ stands for the length of θ.

let us extend δ on $Q \times \Sigma^* \times Q$, denoted also δ, where $q, q' \in Q$ and $\theta \in \Sigma^*$, in the following ways:

(1) $\delta(q, \varepsilon, q') = \begin{cases} 1, & q' = q, \\ 0, & q \neq q', \end{cases}$

(2) $\forall \theta = x_1 x_2 \cdots x_n \in \Sigma^*$,
$$\delta(q, \theta, q') = \bigvee_{q_1, q_2, \cdots, q_n \in Q} [\delta(q, x_1, q_1) \otimes \delta(q_1, x_2, q_2) \otimes \cdots \otimes \delta(q_n, x_n, q')].$$

Definition 2.2([16]): A deterministic lattice-valued fuzzy finite automaton($DLFA$) is a 5-tuple, $\mathscr{L} = (Q, \Sigma, \delta, q_0, F)$, such that $\delta : Q \times \Sigma \longrightarrow Q$, $q_0 \in Q$, $F : Q \longrightarrow L$.

Any L-fuzzy subset in Σ^* is called an L-language on Σ. An L-language accepted or recognized by an LFA $\mathscr{A} = (Q, \Sigma, \delta, I, F)$, denote as $|\mathscr{A}| : \Sigma^* \longrightarrow L$, which is expressed in the form

$$|\mathscr{A}|(\theta) = \bigvee_{q, q' \in Q} [I(q) \otimes \delta(q, \theta, q') \otimes F(q')]$$

for any $\theta \in \Sigma^*$. An L-language which is accepted by an LFA is called an LFA-regular language. Let L_{LFA} denote all LFA-regular language.

An L-language which is accepted by an $DLFA$ is called an $DLFA$-regular language. Let L_{DLFA} denote all $DLFA$-regular languages.

Let $\mathscr{A} = (Q, \Sigma, \delta, I, F)$ be an LFA. For any $q \in Q$, $u \in \Sigma^*$, We define some special L-language $R, R_u : Q \longrightarrow L$, $|\mathscr{A}|(q) : \Sigma^* \longrightarrow L$ as follows: $\forall\ q \in Q$, $u \in \Sigma^*$,

$$R(q) = \bigvee_{q' \in Q, v \in \Sigma^*} [I(q') \otimes \delta(q', v, q)],$$
$$R_u(q) = \bigvee_{q' \in Q} [I(q') \otimes \delta(q', u, q)],$$
$$|\mathscr{A}|(q)(\theta) = \bigvee_{q' \in Q} [\delta(q, \theta, q') \otimes F(q')].$$

For any $q \in Q$, $u \in \Sigma^*$, if $R(q) > 0$, we say that q is a reachable state. If $R_u(q) > 0$, we say that q is a reachable state by word u. If there exists $\theta \in \Sigma^*$, such that $|\mathscr{A}|(q)(\theta) > 0$, we say that q is a accessible state. If q is reachable and accessible, then we say q is trimmed. We say an LFA is trimmed if its all states are trimmed.

III. LRFA AUTOMATA

Definition 3.1([1]): Let f be an L-language over Σ^*, u is a word in Σ^*. The Lattice-valued residual language, denoted as L-residual language, of f with regard to u is defined as

$$u^{-1}f(\theta) = f(u\theta), \quad \forall \theta \in \Sigma^*.$$

Definition 3.2([1]): A lattice-valued fuzzy residual finite automaton($LRFA$) is an LFA $\mathscr{A} = (Q, \Sigma, \delta, I, F)$ such that, for any state $q \in Q$, $|\mathscr{A}|(q)$ is an L-residual language of $|\mathscr{A}|$. More formally, $\forall q \in Q$, $\exists u \in \Sigma^*$, such that $|\mathscr{A}|(q) = u^{-1}|\mathscr{A}|$.

Example 3.1: Let $\mathscr{L}$ be the Gödel structure, $\Sigma = \{a, b\}$. We denote $\frac{r}{\theta}$ ($r \in L, \theta \in \Sigma^*$) as an Lattice language f,ie. $f(\theta) = r$. Let us consider the following LFA.

(1) $\mathscr{A}_1$ in Fig.1 is an LFA, $|\mathscr{A}_1| = \frac{0.5}{a}$, $|\mathscr{A}_1|(q_1) = \frac{0.5}{a}$, $|\mathscr{A}_1|(q_2) = 1$, $a^{-1}|\mathscr{A}_1| = \frac{0.5}{\varepsilon}$, $\varepsilon^{-1}|\mathscr{A}_1| = \frac{0.5}{a}$. Obviously, there does not exist $u \in \Sigma^*$, s.t $u^{-1}|\mathscr{A}_1| = |\mathscr{A}_1|(q_2)$, hence, $\mathscr{A}_1$ isn't an $LRFA$.

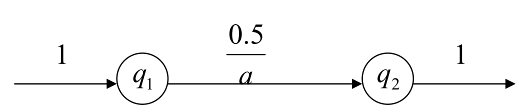

Fig. 1. $\mathscr{A}_1$ isn't an $LRFA$.

(2) $\mathscr{A}_2$ in Fig.2 is an LFA, we have $\mathscr{A}_2$ is an $LRFA$ by Definition 3.2.

An L-language is accepted by an $LRFA$ is called an $LRFA$-regular language. Let L_{LRFA} denote the set of all $LRFA$-regular languages. We can easily prove that an $LRFA$-regular language is also an LFA -regular language.

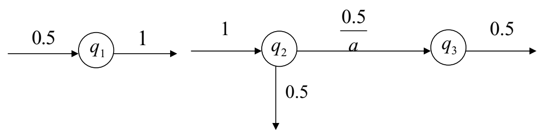

Fig. 2. $\mathscr{A}_2$ is an $LRFA$.

Let f be an L-language, $\lambda \in L$ and let A, B be two sets of L-languages. We define $\lambda \otimes f$ as $\lambda \otimes f(\theta) = \lambda \otimes (f(\theta))(\forall \theta \in \Sigma^*)$. If there exist $\{\lambda_i\}_{i\in I} \subseteq L$, $\{f_i\}_{i\in I} \subseteq A$, such that $f \neq f_i$ for any $i \in I$ and such that $f = \bigcup_{i\in I} \lambda_i \otimes f_i$, then we say f is linear generated by A. We say B be linearly generated by A if every g in B as linear generated by A.

Definition 3.3: Let f is an L-language, we define $Res(f) = \{u^{-1}f | u \in \Sigma^*\}$ is the set of all L-residual languages of f. For any $u^{-1}f \in Res(f)$, if there exists a finite subset C of $Res(f)$ and $u^{-1}f \notin C$ such that $u^{-1}f$ is linear generated by C, then we say $u^{-1}f$ is a composed L-residual language; otherwise, we say $u^{-1}f$ is an irreducible L-residual language. We say a state q is a composed state if the L-residual language $|\mathscr{A}|(q)$ is a composed L-residual language; otherwise, we say q is an irreducible state.

Theorem 3.1: Let $\mathscr{A} = (Q, \Sigma, \delta, I, F)$ be an $LRFA$. For any irreducible L-residual language $u^{-1}|\mathscr{A}|$, there exists state $q \in Q$, such that $u^{-1}|\mathscr{A}| = |\mathscr{A}|(q)$.

Proof: Let $Q_0 = \{q | R_u(q) > 0\}$, then we have $u^{-1}|\mathscr{A}| = \bigvee_{q'\in Q_0} [R_u(q') \otimes |\mathscr{A}|(q')]$. Since $\mathscr{A}$ is an $LRFA$, then, for any state $q' \in Q_0$, there exists $v_{q'} \in \Sigma^*$, such that $v_{q'}^{-1}|\mathscr{A}| = |\mathscr{A}|(q')$. Hence, we have $u^{-1}|\mathscr{A}| = \bigvee_{q'\in Q_0} [R_u(q') \otimes |\mathscr{A}|(q')]$= $\bigvee_{q'\in Q} [(R_u(q') \otimes v_{q'}^{-1}|\mathscr{A}|]$. Since $u^{-1}|\mathscr{A}|$ is irreducible, then there exists $v_q \in \Sigma^*$, such that $u^{-1}|\mathscr{A}| = v_q^{-1}|\mathscr{A}|$, that is $u^{-1}|\mathscr{A}| = v_q^{-1} |\mathscr{A}| = |\mathscr{A}|(q)$. Therefore the conclusion hold. □

IV. Saturation and Reduction operation

In this section, we will define two operators. the first is saturation operator which will add initial and transition functions in an automaton without modifying the language it recognizes. and We can get a unique LFA which has a largest transition and initial functions. the second is reduction operation which may delete some states in an LFA without changing the language it recognizes.

Definition 4.1: Let $\mathscr{A} = (Q, \Sigma, \delta, I, F)$ be an LFA, for any $q, q_1, q_2 \in Q$, $u \in \Sigma^*$, we define two sets $G^I = \{\theta \in \Sigma^* || \mathscr{A}|(q)(\theta) > |\mathscr{A}|(\theta) > 0\}$, $G^\delta = \{\theta \in \Sigma^* || \mathscr{A}|(q_2)(\theta) > u^{-1}|\mathscr{A}|(q_1)(\theta) > 0\}$. The saturated of $\mathscr{A}$ is the LFA $\mathscr{A}^s = (Q, \Sigma, \delta^s, I^s, F)$, where

(1) the initial function I is defined as

$$\begin{cases} 1, & if \quad G^I = \emptyset, \\ \bigwedge_{\theta\in G^I} [|\mathscr{A}|(q)(\theta) \to |\mathscr{A}|(\theta)], & if \quad G^I \neq \emptyset, \end{cases}$$

(2) the transition function $\delta^s(q_1, x, q_2)$ is defined as

$$\begin{cases} 1, & if \quad G^\delta = \emptyset, \\ \bigwedge_{\theta\in G^\delta} [f_{\mathscr{A},q_2}(\theta) \to x^{-1}|\mathscr{A}|(q_1)(\theta)], & if \quad G^\delta \neq \emptyset. \end{cases}$$

If $\mathscr{A} = \mathscr{A}^s$, then we say $\mathscr{A}$ is saturated.

Proposition 4.1: Let $\mathscr{A}$ be an LFA, $\mathscr{A}^s$ be its saturated, then $\mathscr{A}$ and $\mathscr{A}^s$ are state-equivalent, i,e: $\forall q \in Q$,$|\mathscr{A}|(q) = |\mathscr{A}^s|(q)$, and $|\mathscr{A}| = |\mathscr{A}^s|$.

Proof: For any $q, q' \in Q$, $x \in \Sigma$. If $|\mathscr{A}|(q') \leq x^{-1}|\mathscr{A}|(q)$, then $\delta^s(q, x, q') = 1 \geq \delta(q, x, q')$; otherwise, for any $\omega \in G^\delta$, we have

$$\delta(q, x, q') \otimes |\mathscr{A}|(q')(\omega) \leq x^{-1}|\mathscr{A}|(q)(\omega),$$

i.e.,

$$\begin{aligned} \delta(q, x, q') &\leq \bigvee_{\omega\in G^\delta} [|\mathscr{A}|(q')(\omega) \to x^{-1}|\mathscr{A}|(q)(\omega)] \\ &= \delta^s(q, x, q'), \end{aligned}$$

so, $\delta \leq \delta^s$. Similarly, we have $I \leq I^s$. This shows $|\mathscr{A}|(q) \leq |\mathscr{A}^s|(q)$ and $|\mathscr{A}| \leq |\mathscr{A}^s|$.

We prove $|\mathscr{A}^s|(q)(\omega) \leq |\mathscr{A}|(q)(\omega)$ and $|\mathscr{A}^s| \leq |\mathscr{A}|$ for any $\omega \in \Sigma^*$ by induction.

When $\omega = \varepsilon$, clearly, $f_{\mathscr{A},q}(\varepsilon) = F(q) = |\mathscr{A}^s|(q)(\varepsilon)$. Assume now that for any word ω of $|\omega| \leq n$, $|\mathscr{A}|(q)(\omega) \geq |\mathscr{A}^s|(q)(\omega)$. Let $x \in \Sigma^*$,

(1) when $G^\delta = \emptyset$ or $\omega \notin G^\delta$, we have

$$\begin{aligned} |\mathscr{A}^s|(q)(x\omega) &= \bigvee_{q'\in Q} [\delta^s(q, x, q') \otimes |\mathscr{A}|(q')(\omega)] \\ &\leq |\mathscr{A}|(q)(x\omega) \end{aligned}$$

(2) when $G^\delta \neq \emptyset$ and $\omega \in G^\delta$, we have

$$\begin{aligned} &|\mathscr{A}^s|(q)(x\omega) \\ &= \bigvee_{q'\in Q} [\delta^s(q, x, q') \otimes |\mathscr{A}|(q')(\omega)] \\ &= \bigvee_{q'\in Q} \bigwedge_{\theta\in G^\delta} [|\mathscr{A}|(q')(\theta) \to x^{-1}|\mathscr{A}|(q)(\theta)] \otimes |\mathscr{A}|(q')(\omega) \\ &\leq \bigvee_{q'\in Q} [|\mathscr{A}|(q')(\omega) \to x^{-1}|\mathscr{A}|(q)(\omega)] \otimes |\mathscr{A}|(q')(\omega) \\ &= x^{-1}|\mathscr{A}|(q)(\omega) = |\mathscr{A}|(q)(x\omega). \end{aligned}$$

Hence, we have $|\mathscr{A}^s|(q)\ (x\omega) \leq |\mathscr{A}|(q)(x\omega)$. Similarly, we can prove $|\mathscr{A}^s| \leq |\mathscr{A}|$. Therefore we have $|\mathscr{A}|(q) = |\mathscr{A}^s|(q)$ and $|\mathscr{A}| = |\mathscr{A}^s|$. □

Proposition 4.2: Let $\mathscr{A}_1$, $\mathscr{A}_2$ are two state-equivalent LFAs, i.e.,$|\mathscr{A}_1| = |\mathscr{A}_2|$, $|\mathscr{A}_1|(q) = |\mathscr{A}_2|(q)$ $(\forall q \in Q)$, then we have

$\mathscr{A}_1^s = \mathscr{A}_2^s$.

Proof: By the definition of saturation operator, for any $q \in Q$, $|\mathscr{A}_1|(q) \le |\mathscr{A}_1|$ if and only if $|\mathscr{A}_2|(q) \le |\mathscr{A}_2|$, that is, $I_1^s(q) = I_2^s(q) = 1$; otherwise, $G_1^I = G_2^I$ is not empty, hence, we have

$$\begin{aligned} I_1^s(q) = & \bigwedge_{\theta \in G_1^I} |\mathscr{A}_1|(q)(\theta) \to |\mathscr{A}_1|(\theta)] \\ = & \bigwedge_{\theta \in G_2^I} [|\mathscr{A}_2|(q)(\theta) \to |\mathscr{A}_2|(\theta)] = I_2^s(q). \end{aligned}$$

Therefore, $I_1^s = I_2^s$.

By the definition of saturation operator, for any $q_1, q_2 \in Q$, $x \in \Sigma$, $|\mathscr{A}_1|(q_2) \le x^{-1}\ |\mathscr{A}_1|(q_1)$ if and only if $|\mathscr{A}_2|(q_2) \le x^{-1}\ |\mathscr{A}_2|(q_1)$, hence, $\delta_1^s\ (q_1, x, q_2) = \delta_2^s\ (q_1, x, q_2)$; otherwise, $G_1^\delta = G_2^\delta$ is not empty, that is, we have

$$\begin{aligned} \delta_1^s(q_1, x, q_2) = & \bigwedge_{\theta \in G_1^\delta} [|\mathscr{A}_1|(q_1)(\theta) \to x^{-1}|\mathscr{A}_1|(q_2)(\theta)] \\ = & \bigwedge_{\theta \in G_2^\delta} [|\mathscr{A}_2|(q_1)(\theta) \to x^{-1}|\mathscr{A}_2|(q_2)(\theta)] \\ = & \ \delta_2^s(q_1, x, q_2), \end{aligned}$$

hence, we have $\delta_1^s = \delta_2^s$. Since $|\mathscr{A}_1|(q) = |\mathscr{A}_2|(q)\ (\forall q \in Q)$, then we have $F_1(q) = |\mathscr{A}_1|(q)(\varepsilon) = |\mathscr{A}_2|(q)\ (\varepsilon) = F_2(q)$, hence, $F_1 = F_2$.

Therefore $\mathscr{A}_1^s = \mathscr{A}_2^s$. □

Theorem 4.1: Let $\mathscr{A}$ be an LFA, $\mathscr{A}^s$ be its saturated, then we have $\mathscr{A}^s = (\mathscr{A}^s)^s$.

Proof: For any $q \in Q$, we have $|\mathscr{A}|(q) = |\mathscr{A}^s|(q)$ and $|\mathscr{A}| = |\mathscr{A}^s|$ by Proposition 4.1, hence, we have $\mathscr{A}^s = (\mathscr{A}^s)^s$ by Proposition 4.2. □

Theorem 4.2: If $\mathscr{A}$ is an $LRFA$, then $\mathscr{A}^s$ is also an $LRFA$.

Proof: By definition of $LRFA$, for any $q \in Q$, there exists $u \in \Sigma^*$, such that $|\mathscr{A}|(q) = u^{-1}|\mathscr{A}|$. Since $|\mathscr{A}|(q) = |\mathscr{A}^s|(q)$, $|\mathscr{A}^s| = |\mathscr{A}|$, we have $|\mathscr{A}^s|(q) = |\mathscr{A}|(q) = u^{-1}|\mathscr{A}| = u^{-1}|\mathscr{A}^s|$. Hence, $\mathscr{A}^s$ is also an $LRFA$. □

Definition 4.2: Let $\mathscr{A} = (Q, \Sigma, \delta, I, F)$ be an LFA, for any $q \in Q$, we define a set $N(q) = \{q' \in Q \backslash \{q\} | Supp(|\mathscr{A}|(q')) \subseteq Supp(|\mathscr{A}|(q))\}$.

For any $q' \in N(q)$, if there exists $\lambda_{q'} \in L$, such that $|\mathscr{A}|(q) = \bigvee_{q' \in N(q)} \lambda_{q'} |\mathscr{A}|(q')$, we say q is erasable. When q is erasable, we define reduction operator $\mathscr{A}' = \phi(\mathscr{A}, q) = (Q', \Sigma, \delta', I', F')$, where

(1) $Q' = Q \backslash \{q\}$,

(2) the initial function is defined as: for any $q' \in Q'$,

$$I(q') = \begin{cases} I(q'), & if \quad q' \notin N(q), \\ I(q') \vee \lambda_{q'} I(q), & if \quad q' \in N(q), \end{cases}$$

(3) For any $q_1, q_2 \in Q', x \in \Sigma$, the transition function $\delta'(q_1, x, q_2)$ is defined as:

$$= \begin{cases} \delta(q_1, x, q_2), & if \quad q_2 \notin N(q), \\ \delta(q_1, x, q_2) \vee \lambda_{q'} \delta(q_1, x, q), & if \quad q_2 \in N(q), \end{cases}$$

(4) $F' = F$.

If q is not erasable, we define $\phi(\mathscr{A}, q) = \mathscr{A}$. If $LFA\ \mathscr{A}$ has no erasable state, we say that $LFA\ \mathscr{A}$ is reduced.

Proposition 4.3: Let $\mathscr{A}$ be an LFA, for any $q \in Q$, if $\mathscr{A}' = \phi(\mathscr{A}, q)$, then for any $q' \in Q'$, we have $|\mathscr{A}'|(q') = |\mathscr{A}|(q')$, and $|\mathscr{A}'| = |\mathscr{A}|$.

Proof: When q is not erasable, the conclusion is clear.

When q is erasable, let $\mathscr{A} = (Q, \Sigma, \delta, I, F)$ be an LFA, $\phi(\mathscr{A}, q) = \mathscr{A}' = (Q', \Sigma, \delta', I', F)$. We prove $\forall q' \in Q'$, $|\mathscr{A}', q'| = |\mathscr{A}, q'|$ by induction on ω.

If $\omega = \varepsilon$, then $|\mathscr{A}'|(q')(\varepsilon) = F(q') = |\mathscr{A}|(q')(\varepsilon)$.

When $\omega' = x\omega$, we have

$$\begin{aligned} & |\mathscr{A}'|(q')(x\omega) \\ = & \bigvee_{q'' \in Q'} [\delta'(q', x, q'') \otimes |\mathscr{A}', q''|(\omega)] \\ = & \bigvee_{q'' \in Q' \backslash N(q)} [\delta(q', x, q'') \otimes |\mathscr{A}'|(q'')(\omega)] \\ \vee & \bigvee_{q'' \in N(q)} [\delta'(q', x, q'') \otimes |\mathscr{A}'|(q'')(\omega)] \\ = & \bigvee_{q'' \in Q' \backslash N(q)} [\delta'(q', x, q'') \otimes |\mathscr{A}'|(q'')(\omega)] \\ \vee & \bigvee_{q'' \in N(q)} [(\delta(q', x, q'') \vee \lambda_{q''} \delta(q', x, q)) \otimes |\mathscr{A}'|(q'')(\omega)] \\ = & \bigvee_{q'' \in Q' \backslash N(q)} [\delta(q', x, q'') \otimes |\mathscr{A}', q''|(\omega)] \end{aligned}$$

$$\begin{aligned} \vee & \bigvee_{q'' \in N(q)} [\delta(q', x, q'') \otimes |\mathscr{A}'|(q'')(\omega)] \\ \vee & \bigvee_{q'' \in N(q)} [\delta(q', x, q) \otimes \lambda_{q''} |\mathscr{A}'|(q'')(\omega)] \\ = & \bigvee_{q'' \in Q' \backslash N(q)} [\delta(q', x, q'') \otimes |\mathscr{A}|(q'')(\omega)] \\ \vee & \bigvee_{q'' \in N(q)} [\delta(q', x, q'') \otimes |\mathscr{A}|(q'')(\omega)] \\ \vee & [\delta(q', x, q) \otimes |\mathscr{A}|(q)(\omega)] \\ = & \bigvee_{q' \in Q} [\delta(q', x, q'') \otimes |\mathscr{A}|(q'')(\omega)] = |\mathscr{A}|(q')(x\omega). \end{aligned}$$

Hence, we have $|\mathscr{A}'|(q') = |\mathscr{A}|(q')$, $\forall q' \in Q'$.

By the definition of L-language accepted by an LFA, we have

$$\begin{aligned}|\mathscr{A}'| &= \bigvee_{q'\in Q'} [I'(q')\otimes|\mathscr{A}'|(q')]\\
&= \bigvee_{q'\in Q'\setminus N(q)} [I'(q')\otimes|\mathscr{A}'|(q')] \vee \bigvee_{q'\in N(q)} [I'(q')\otimes|\mathscr{A}'|(q')]\\
&= \bigvee_{q'\in Q'\setminus N(q)} [I(q')\otimes\mathscr{A}|(q')]\\
&\vee \bigvee_{q'\in N(q)} [I(q')\otimes|\mathscr{A}|(q')] \vee \bigvee_{q'\in N(q)} [I(q)\otimes\lambda_{q''}|\mathscr{A}|(q')]\\
&= \bigvee_{q'\in Q'\setminus N(q)} [I(q')\otimes|\mathscr{A}|(q')]\\
&\vee \bigvee_{q'\in N(q)} [I(q')\otimes|\mathscr{A}|(q')] \vee [I(q)\otimes|\mathscr{A}|(q)] = |\mathscr{A}|.\end{aligned}$$

Hence, $|\mathscr{A}'| = |\mathscr{A}|$. The proof is completed. □

Theorem 4.3: Let $\mathscr{A} = (Q, \Sigma\ , \delta, I, F)$ be an $LRFA$. For any state $q \in Q$, $\mathscr{A}' = \phi(A, q)$ is also an $LRFA$.

Proof: For any state $q \in Q$, there exists $u \in \Sigma^*$, such that $|\mathscr{A}|(q) = u^{-1}|\mathscr{A}|$. By Proposition 4.3, we have $|\mathscr{A}'|(q) = |\mathscr{A}|(q)$,$|\mathscr{A}'| = |\mathscr{A}|$, that is, $|\mathscr{A}'|(q) = u^{-1}|\mathscr{A}'|$. Therefore, $\mathscr{A}'$ is an $LRFA$. □

Theorem 4.4: Let $\mathscr{A} = (Q, \Sigma, \delta, I, F)$ be an $LRFA$. For any $q \in Q$, if $\mathscr{A}$ is saturated, then $\phi(\mathscr{A}, q)$ is saturated.

Proof: If $q' \notin N(q)$, then $I'(q') = I(q')$. Since A is saturated, we have $I^s(q') = I(q')$, hence, $(I')^s(q') = I^s(q') = I(q') = I'(q')$, that be $(I')^s(q') = I'(q')$. If $q' \in N(q)$, then $I'(q') = I(q')\vee\ \lambda_{q'}\ I(q)$. $(I')^s(q') = [I(q') \vee \lambda_{q'}\ I(q)]^s = I^s(q') \vee \lambda_{q'} I^s(q) = I(q') \vee \lambda_{q'}\ I(q) = I'(q')$. Therefore, $(I')^s = I'$.

If $q_2 \notin N(q)$, since $\mathscr{A}$ is saturated, we have $(\delta')^s(q_1, x, q_2) = \delta^s(q_1, x, q_2) = \delta(q_1, x, q_2) = \delta'(q_1, x, q_2)$. If $q_2 \in N(q)$, since $\mathscr{A}$ is saturated, we have $(\delta')^s(q_1, x, q_2) = [\delta(q_1, x, q_2) \vee \lambda_{q_2}\ \delta\ (q_1, x, q)]^s = \delta^s(q_1, x, q_2) \vee \lambda_{q_2}\ \delta^s(q_1, x, q) = \delta'\ (q_1, x, q_2)$, hence, we have $(\delta')^s = \delta'$. Therefore, $(\mathscr{A}')^s = \mathscr{A}'$, i.e., $\phi(\mathscr{A}, q)$ is saturated. □

Theorem 4.5: Let $\mathscr{A} = (Q, \Sigma, \delta, I, F)$ be an LFA. For any state $q \in Q$, we have $[\phi(\mathscr{A}, q)]^s = \phi\ (\mathscr{A}^s, q)$.

Proof: By definitions of saturation and reduction operator, $\mathscr{A}'$ and $(\mathscr{A}^s)'$ have the same states set $Q' = Q\setminus\{q\}$, for any $q \in Q$. By Theorem 4.1, we have $((\mathscr{A}^s)')^s = (\mathscr{A}^s)'$. By Proposition 4.1, 4.3, we have $|\mathscr{A}'| = |(\mathscr{A}^s)'|$, $|\mathscr{A}'|(q') = |(\mathscr{A}^s)'|(q')$, $\forall q' \in Q'$. By Proposition 4.2, we have $(\mathscr{A}')^s = ((\mathscr{A}^s)')^s = (\mathscr{A}^s)'$. Hence, we have $[\phi(\mathscr{A}, q)]^s = \phi(\mathscr{A}^s, q)$. □

V. Canonical $LRFA$

Definition 5.1: Let f be an $LRFA$-regular language. We define the canonical $LRFA$ of f as the following way $\mathscr{A} = (Q, \Sigma, \delta, I, F)$, where

(1) Σ is the alphabet of f,
(2) Q is the set of irreducible L-residual languages, i.e., $Q = \{u^{-1}f | u^{-1}f$ is a irreducible $\}$,
(3) we define a set $G = \{\theta \in \Sigma^* | u^{-1}f(\theta) > f(\theta) > 0\}$, the initial function $I(u^{-1}f)$ is defined as

$$\begin{cases} 1, & if \quad G = \emptyset,\\ \bigwedge_{\theta\in G} [u^{-1}f(\theta) \to f(\theta)], & if \quad G \neq \emptyset,\end{cases}$$

(4) the final function is defined as $F(u^{-1}f) = u^{-1}f(\varepsilon)$,
(5) we define a set $G = \{\theta \in \Sigma^* | v^{-1}f(\theta) > (ux)^{-1}f(\theta) > 0\}$, the transition function $\delta^s(u^{-1}f, x, v^{-1}f)$ is defined as

$$\begin{cases} 1, & if \quad G = \emptyset,\\ \bigwedge_{\theta\in G} [v^{-1}f(\theta) \to (ux)^{-1}f(\theta)], & if \quad G \neq \emptyset.\end{cases}$$

The above definition also gives a method for constructing the canonical $LRFA$ from a given L-regular language.

Example 5.1: Let $\mathscr{L}$ be the Gödel structure, Consider L-language $f = \frac{0.5}{ab^*} + \frac{1}{ba^*}$. $\varepsilon^{-1}f = f$, $a^{-1}f = \frac{0.5}{b^*}$, $b^{-1}f = \frac{1}{a^*}$ are irreducible residual L-languages, there exists $\{q_1, q_2, q_3\}$, such that $|\mathscr{A}|(q_1) = \varepsilon^{-1}f$, $|\mathscr{A}|(q_2) = a^{-1}f$, $|\mathscr{A}|(q_3) = b^{-1}f$. The canonical of f is defined as shown in Fig.3.

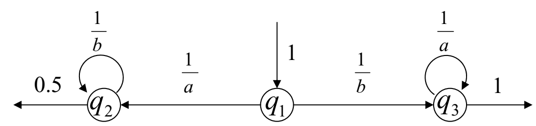

Fig. 3. $\mathscr{A}_3$ is the canonical $LRFA$ recognizing f.

Theorem 5.1: Let f be an $LRFA$-language. If $\mathscr{A} = (Q, \Sigma, \delta, I, F)$ is a reduced and saturated $LRFA$ recognizing f, then $\mathscr{A}$ is the canonical $LRFA$ of f.

Proof: As $\mathscr{A}$ is an $LRFA$, every irreducible L-residual language $u^{-1}f$ can be defined as an L-language $|\mathscr{A}|(q)$ associated with some states $q \in Q$. As there are no erasable state in Q. For any state q, $|\mathscr{A}|(q)$ is a irreducible L-residual language and distinct states define distinct L-residual languages. Since $\mathscr{A}$ is saturated, then it has the same initial and transition functions with the canonical $LRFA$. □

Theorem 5.2: The canonical $LRFA$ of an $LRFA$-regular language f is an $LRFA$ which recognizes f and which is minimal $LRFA$ regarding the number of states.

Proof: Let $\mathscr{A}_0$, $\mathscr{A}_1$, $\cdots$, $\mathscr{A}_n$ be a sequence of $LRFA$ such that for any index $i \geq 1$, there exists a state q_i of $\mathscr{A}_i$, such that $\mathscr{A}_i = \phi(\mathscr{A}_{i-1}, q_i)$. Theorem 4.5 and 5.5 show that $\mathscr{A}_n$ is the canonical $LRFA$ of the language recognized by $\mathscr{A}_0$ if $\mathscr{A}_0$ is a saturated $LRFA$ and $\mathscr{A}_n$ is reduced. So the canonical $LRFA$ can be obtained from any $LRFA$ that recognizes f using saturation and reduction operators. Theorem 3.1 implies that it has a minimal number of states. □

Theorem 5.3: The canonical $LRFA$ of an $LRFA$-regular language f is the unique $LRFA$ that has a largest transition and initial functions among the set of $LRFA$ which have a minimal number of states.

Proof: Let $\mathscr{A} = (Q, \Sigma, \delta, I, F)$ be the canonical $LRFA$ of an $LRFA$-language f and let $\mathscr{A}' = (Q', \Sigma, \delta', I', F')$ be an $LRFA$ which has a minimal number of states. Hence, $\mathscr{A}'$ is reduced. From Theorem 5.2, the saturated automaton of $\mathscr{A}'$ is $\mathscr{A}$. Therefore, $\mathscr{A}'$ has at most as large transition and initial functions as $\mathscr{A}$. □

Theorem 5.4: $L_{DLFA} \subsetneq L_{LRFA} \subsetneq L_{LFA}$.

We have the conclusion of $L_{DLFA} \subseteq L_{LRFA} \subseteq L_{LFA}$ in section 3, we can proof $L_{DLFA} \subsetneq L_{LRFA} \subsetneq L_{LFA}$ by example as follows.

Example5.2: Let $\mathscr{L}$ be the Gödel structure.

(1) $f = (\frac{0.5}{a})^*$ is an L-language, $\{(a^n)^{-1}f | n \in \mathbb{N} \cup \{0\}\}$ is the set of irreducible L-residual languages of f. $(a^n)^{-1}f = (0.5)^n \otimes \varepsilon^{-1}f$ for any $n \in \mathbb{N} \cup \{0\}$,. Hence, $\varepsilon^{-1}f$ is the unique irreducible L-residual language of f, the canonical $LRFA$ of f has only a state. we can build the canonical $LRFA$(Fig4) of f. f can't be recognized by a $DLFA$. therefore $L_{DLFA} \subsetneq L_{LRFA}$.

(2) $f = (\frac{0.5}{a})^*(\frac{0.5}{b})^*$ is an L-language, $\mathscr{A}_5$(Fig.5) is an LFA recognized f . Then f is LFA-language. $g = \sum_{k \geq 0}(\dfrac{0.5}{a})^k(\dfrac{0.5}{b})^k$ is the set of subclass language of f, for any $n \in \mathbb{N} \cup \{0\}$, $(u^n)^{-1}g = \sum_{k \geq n} 0.5^n(\dfrac{0.5}{a})^{k-n}(\dfrac{0.5}{b})^k$ is L-residual language of g with regard to u^n. hence, $\{u^n)^{-1}g | n \in \mathbb{N} \cup \{0\}\}$ is the set of L-residual language of g, but $\{u^n)^{-1}g | n \in \mathbb{N} \cup \{0\}\}$ is infinite. so, g can't be recognized by an $LRFA$. Therefore, $L_{LRFA} \subsetneq L_{LFA}$.

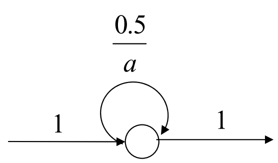

Fig. 4. $\mathscr{A}_4$(Fig.4) is an LFA recognized f.

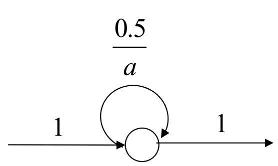

Fig. 5. $\mathscr{A}_5$(Fig.5) is an LFA recognized f

In [19], Li gives an algorithm to transform an LFA into an equivalent $DLFA$ in some more general frames. Consider an LFA $\mathscr{A} = (Q, \Sigma, \delta, I, F)$, where $Q = \{q_1, q_2, \cdots, q_n\}$. The following algorithm build a $DLFA$ $\mathscr{A}' = (Q', \Sigma, \delta', q_0', F')$ which is equivalent to $\mathscr{A}$. For more details, refer to [19, 15].

Algorithm 1.

Step1: Let $M = \{\delta(q_i, u, q_j) | 1 \leq i, j \leq n, u \in \Sigma^*\} \cup \{I(q) : q \in Q\}$. We generate a finite sublattice $M_1 = \langle M \rangle = \{\vee Z | Z$ is a finite subset of $X\}$, where $X = \{a_1 \otimes \cdots a_k | \{a_1, \cdots, a_k\}$ is a finite subset of $M\}$, we get a finite sublattice M_1 of L.

Step2: Let $Q' = \{(a_1, a_2, \cdots, a_n) | a_i \in M_1\}$, define $\delta' : Q \times \Sigma \to Q$ as $\delta'((a_1, a_2, \cdots, a_n), u) = (b_1, b_2, \cdots, b_n)$, where $b_i = \bigvee_{j=1}^{n} [a_j \otimes \delta(q_j, u, q_i)].q_0' = (I(q_1), I(q_2), \cdots, I(q_n))$ and $F' : Q' \to L$ as $F_1(a_1, a_2, \cdots, a_n) = \bigvee_{i=1}^{n} [a_i \otimes F(q_i)]$.

In general, the set X in Algorithm 1 is not a finite set, and thus Q' is not finite. To guarantee that the $DLFA$ constructed in Algorithm 1 is finite, we require the algebra $(L, \otimes)$ is locally finite, i.e., the subalgebra generated by any finite set of $(L, \otimes)$ is also finite ([15]).

For an LFA $\mathscr{A}$, we can have the following algorithm transform it into the canonical $LRFA$ of it based on Algorithm 1.

Algorithm 2.

Step1: We can build a $DLFA$ $\mathscr{B}$ from $\mathscr{A}$ by Algorithm 1. Clearly, $DLFA$ $\mathscr{B}$ is also an $LRFA$.

Step2: We obtain its saturated $DLFA$ $\mathscr{B}^s$ by the saturation operator.

Step3: We reduce its all erasable states, we can obtain its saturated and reduced $LRFA$ $\mathscr{C}$. Clearly, $\mathscr{C}$ is the canonical $LRFA$ of $\mathscr{A}$.

Example 5.3: Let $\mathscr{L}$ be the Gödel structure, we consider the LFA $\mathscr{A}_{10}$(Fig.6) recognizing the $LRFA$-language $f = \frac{0.5}{a^*} + \frac{0.7}{bb^*} + \frac{0.7}{ab^*}$. Since $\mathscr{A}_6$ has two irreducible residual L-languages $\varepsilon^{-1}f = f$, $b^{-1}f = \frac{0.7}{b^*}$, the canonical $LRFA$ recognizing $\mathscr{A}_6$ has two states. We build the canonical $LRFA$ from LFA $\mathscr{A}_6$ by Algorithm 2.

Step1: We build a $DLFA$ by Algorithm 1, reduce all states which are not reachable or are not accessible, then we can obtain a trimmed $DLFA$ $\mathscr{A}_7$(Fig.7).

Step2: We can obtain its saturated and reduced $LRFA$ $\mathscr{A}_8$(Fig.8) by saturation and reduction operator, and $\mathscr{A}_8$ is the canonical $LRFA$ recognizing the $LRFA$-language f.

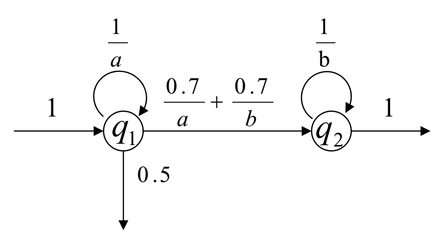

Fig. 6. $\mathscr{A}_6$ is an LFA recognizing the L-language f.

We take the number of states of an automaton as a measure of its size. $LRFA$ is a subclass of LFA and trimmed $DLFA$ are special $LRFA$. The canonical $LRFA$ is the minimal

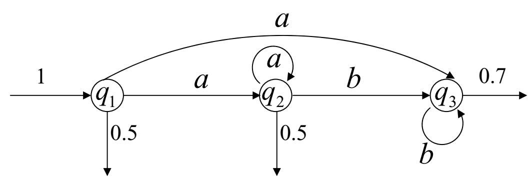

Fig. 7. Fig.11. $\mathscr{A}_7$ is a trimmed $DLFA$ recognizing L-regular language f.

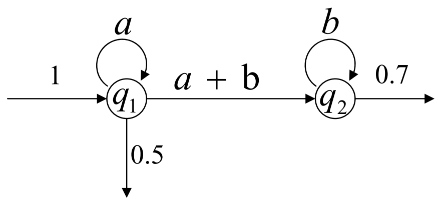

Fig. 8. $\mathscr{A}_8$ is the canonical $LRFA$ recognizing $LRFA$-regular language f.

$LRFA$ recognizing the same L-language. Hence, we have the following proposition.

Theorem 5.6: For a given $LRFA$-regular language f. The canonical $LRFA$ recognizing f has the size of the equivalent minimal $DLFA$ as an upper bound and the size of its equivalent minimal LFA as a lower bound.

VI. Conclusions

In this paper, we introduced the notion of $LRFA$ and $LRFA$-regular language with membership values in a complete residuated lattice. Then we gave the definitions of saturation operator and reduction operator of LFA, obtained closures of two operators in $LRFA$. There is a unique saturated and reduced $LRFA$ for an LFA. In particular, we proved that, for every $LRFA$-regular language f, there exists a unique $LRFA$ which has a minimal number of states and largest initial and transition functions, we call it as canonical $LRFA$, and gave a construction method. The study provides another way for the minimization of states of lattice-valued finite automata.

The next problem is to consider lattice-valued residual finite automata with membership valued in a general lattice-ordered monoid, and further problem is weighted residual finite automaton. In this paper, the definition of saturation and canonical $LRFA$ depended on the structure of implication operator in complete residual lattice, where implication operator was difficult to define in a lattice -ordered monoid or in a semiring. Therefore, we need to search new method to define the saturator and canonical $LRFA$ in the case the truth values taken in a lattice-ordered monoid or in a semiring, which forms another research topic in the further work.

References

[1] F. Denis, A. Lemay, A. Terlutte, Residual finite ftate automata, STACS 2001, 18th Annual Symposium on Theoretical Aspects of Computer Science, number 2010 in Lecture Notes in Computer Science, *Springer Verlag.*, (2001) 144–147.

[2] F. Denis, A. Lemay, A. Terlutte, Learning regular languages using RFSA, In ed., Proceedings of the 12th International Conference on Algorithmic Learning Theory (ALT-01). Number 2225 in Lecture Notes in Computer Science,*Springer Verlag.*,(2001) 348–359.

[3] Y. Esposito, A. Lemay, F. Denis, P. Dupont, Learning probabilistic residual finite state automata, In: ICGI'2002, 6th International Colloquium on Grammatical Inference. LNAI, *Springer Verlag.*, (2002) 77–91.

[4] Y. Esposito, A. Lemay, F. Denis, P. Dupont, Learning probabilistic residual finite state automata, *Springer-Verlag Berlin Heidelberg 2002 Informaticae.*51 (2002) 77–91.

[5] V.M. Glushkov, The abstract theory of automata, *Russian Math.* Surveys 16 (1961) 1–53.

[6] L. Ilie, G. Navarro, S. Yu, On NFA Reductions, *Springer-Verlag Berlin Heidelberg 2004, Karhumäki et al. (Eds.): Theory Is Forever (Salomaa Festschrift).* Springer-Verlag Berlin Heidelberg 2004, Karhumäki et al. (Eds.): Theory Is Forever (Salomaa Festschrift), LNCS 3113 112–124.

[7] L. Ilie, S. Yu, Reducing NFAs by invariant equivalences.*Theoretical Computer Science.*, 306 (2003)373–390.

[8] G. Gramlich, G. Schnitger, Minimizing nfa's and regular expressions, *Journal of Computer and System Sciences.* 73(6)(2007) 908–923.

[9] G. Gramlich, G. Schnitger, Minimizing NFAs and Regular Expressions. STACS 2005, *Lecture Notes in Computer Science.* , 3404(2005) 399–411.

[10] J. M. Champarnaud, F. Coulon, NFA reduction algorithms by means of regular inequalities, *Theoretical Computer Science.*, 327(3) (2004) 241–253.

[11] L.A. Zadeh, Fuzzy sets, *Inform. Control.*, 8 (1965) 338–353.

[12] W. G. Wee, On generalizations of adaptive algorithm and application of the fuzzy sets concept to pattern classification, Ph.D. Thesis, Purdue University, June 1967.

[13] R. Bělohlávek, Determinism and fuzzy automata, *Inform. Scie.*, 142 (2002) 205–209.

[14] L. H. Wu, D. W. Qiu, Automata theory based on complete residuated lattice-valued logic: Reduction and minimization. *Fuzzy Set and Systems.*, 161 (2010) 1635–1656.

[15] Y. M. Li, W. Pedrycz, Fuzzy finite automata and fuzzy regular expressions with membership values in lattice-ordered monoids, *Fuzzy Sets and Systems.*, 156 (2005), 68–92.

[16] Y. M. Li, Finite automata theory with membership values in lattices, *Information Sciences.*, 181 (2011) 1003–1017.

[17] D. W. Qiu, Automata theory based on completed residuated lattice-valued logic (I), *Science in China.*, Ser. F, 44(6) (2001) 419–429.

[18] D. W. Qiu, Characterizations of fuzzy finite automata,*Fuzzy Sets and Systems.* 141 (2004) 391–414.

[19] Y. M. Li, W. Pedrycz, Minimization of lattice finite automata and its application to the decomposition of lattice languages, *Fuzzy Sets and Systems.*, 158 (2007) 1423–1436.

[20] Y. M. Li , Q. Wang, S.J. Li, On quotients of formal power series, arXiv:1203.2236v1 [cs.FL] 10 Mar 2012, 2011, preprint.

[21] H. X. Lei, Y.M. Li. Minimization of states in automata theory based on finite lattice-ordered monoids. *Information Sciences.*, 177 (2007) 1413–1423.

[22] J. Ignjatović, M. Ćirić, S. Bogdanovicé, T. Petkovicé, Myhill-Nerode type theory for fuzzy languages and automata, *Fuzzy Sets and Systems.*, 161 (2010) 1288–1324.

[23] M. Ćirić, A. Stamenković, J. Ignjatović, T. Petkovicé, Fuzzy relation equations and reduction of fuzzy automata. *Journal of Computer and System Sciences.*, 76 (2010) 609–633.

[24] J. Ignjatović, M. Ćirić, S. Bogdanovicé, Determinization of fuzzy automata with membership values in complete residuated lattices. *Information Sciences.*, 178 (2008) 164–180.

[25] M. Ćirić, J. Ignjatović, S. Bogdanović, Fuzzy equivalence relations and their equivalence classes, *Fuzzy Sets and Systems.* 158 (2007) 1295–1313.

[26] M. Ćirić, A. Stamenković, J. Ignjatović, T. Petković, Factorization of fuzzy automata, In: E. Csuhaj-Varju, Z. Ésik (eds.), FCT 2007, *Lecture Notes in Computer Science.* 4639 (2007) 213–225.

[27] Z. Jančić, J. Ignjatović, M. Ćirić, "An improved algorithm for determinization of weighted and fuzzy automata", *Information Sciences.* 181 (2011) 1358–1368.

[28] T. Petković, "Congruences and homomorphisms of fuzzy automata", *Fuzzy Sets and Systems* .157 (2006) 444–458.

[29] R. Bělohlávek, *Fuzzy Relational Systems: Foundations and Principles, Kluwer, New York.*, 2002.
[30] R. Bělohlávek, V. Vychodil, ”Fuzzy Equational Logic”, *Springer, Berlin, Heidelberg.*, 2005.