LATTICE VALUED ALESHIN TYPE AUTOMATA WITH ε-MOVES

A.Jeyanthi[1] V.Kalpana[2]

FACULTY, ANNA UNIVERSITY, REGIONAL OFFICE, MADUAI, TAMILNADU, I NDIA

ABSTRACT:

The extended subset construction of lattice-valued Aleshin type finite automata is introduced, then the equivalences between lattice-valued finite automat , lattice-valued deterministic finite automata and lattice-valued finite automata with e-moves are proved. A simple characterization of lattice-valued languages recognized by lattice-valued finite automata is given, then it is proved that the Kleene theorem holds in the frame of lattice-setting. A minimization algorithm of lattice-valued deterministic finite automata is presented. In particular, the role of the distributive law for the truth valued domain of finite automata is analyzed: the distributive law is not necessary to many constructions of lattice-valued finite automata, but it indeed provides some convenience in simply processing lattice-valued finite automata.

KEYWORDS: lattice-valued Aleshin type lattice-valued deterministic finite automata,

INTRODUCTION: The contribution of this study contains at least three aspects. First, as we just said, lattice-valued finite automata in this study is a common generalization of fuzzy automata and weighted automata. In this respective, the role of the distributive law for the truth valued domain of finite automata is analyzed. It is demonstrated that the distributive law is not necessary to many constructions of lattice-valued finite automata, but it indeed provides some convenience in simply processing latticevalued finite automata. Second, the technique of extended subset construction is introduced, using this technique, the equivalence between lattice-valued finite automata and lattice-valued deterministic finite automata is proved. Some results in [33] were strength, especially, the Kleene theorem in lattice-setting is presented. Third, we give a minimal algorithm of lattice-valued deterministic finite automata.

The content of this paper is arranged as follows. In Section 2, the definition of lattice-valued finite automata is introduced, then the relationship between the extendability of the state transition relation d and the distributive law of the lattice l is exploited. Section 3 introduces the extended subset construction of l-VFA, the determinization of l-VFA is shown. The relationship between extended subset construction and lattice-valued subset construction and the distributive law of l is discussed. Some results on determinization of lattice-valued finite automata in [1] are strengthened. In Section II, some simple characterizations of l-valued regular languages are introduced, the operations on the l-valued regular languages are discussed and Kleene Theorem in lattice setting is established. Section 5 discusses the minimization of l-VDFA, the minimal algorithm of l-VDFA is presented. Some remarks are included in the conclusion part.

II LATTICE VALUED FINITE STATE AUTOMATA

Definition 3.2 (c.f. [22,33,26]). An l-valued automaton with ε-moves (l-VFA$_\varepsilon$ for short) is a five-tuple $A=(Q,\Sigma,\delta,I,F)$ in which all components are the same as in an l-valued automaton (without ε-moves), but the domain of the l-valued transition relation δ is changed to $Q \times (\Sigma \cup \{\varepsilon\}) \times Q$; that is, δ is a mapping from $Q \times (\Sigma \cup \{\varepsilon\}) \times Q$ into l, where ε stands for the empty string of input symbols.

Now let A = (Q,Σ,δ, I, F) be an l-valued automaton with ε-moves. Then the recognizability rec_A is also defined as an l-valued unary predicate over Σ*, and it is given by

$$\omega \in rec_A =_{def} (\exists n \geq 0)(\exists \tau_1 \in \Sigma \cup \{\varepsilon\}) \ldots (\exists \tau_n \in \Sigma \cup \{\varepsilon\}) . (\exists q_0 \in Q) \ldots (\exists q_n \in Q) . (q_0 \in I \wedge q_n \in F \wedge (q_0,\tau_1,q_1) \in \delta \wedge \ldots \wedge (q_{n-1},\tau_n,q_n) \in \delta \wedge \tau_1 \ldots \tau_n = \omega)$$

for all ω ∈ Σ*. The defining equation of rec_A may be rewritten in terms of truth value as follows:

$rec_A(\omega) = V\ \{I(q_0) \wedge \delta\ (q_0,\tau_1,q_1) \wedge \ldots \wedge \delta(q_{n-1},\tau_n,q_n) \wedge F(q) : n \geq 0,\ \tau_1, \ldots, \tau_n \in \Sigma \cup \{\varepsilon\}$ satisfying $\tau_1 \ldots \tau_n = \omega$, and $q_0, \ldots, q_n \in Q\}$.

We shall show that *l*-VAFA and *l*-VAFA$_\in$ are equivalent in the sequel. First, we study a special kind of l-VFA$_\in$ in which transition is crisp, that is , δ is a crisp subset of Q × (Σ ∪ {ε}) × Q. In this case, δ can be seen as a mapping from Q × (Σ ∪ {ε}) to 2^Q.

Let A = (Q,Σ,δ,q_0,F) be an l-VFA$_\in$ with crisp transition and with a unique initial state $q_0 \in Q$, the explicit expression of rec_A can be induced as follows. Frist, we give the extension $\delta^* : 2^Q \times \Sigma^* \to 2^Q$ using the notion of ε-closure. For q ∈ Q, the ε-closure of q, denoted EC(q), is defined as,

EC(q) = {p ∈ Q : there exists n ≥ 0 and $q_0, \ldots, q_n$ satisfying $q_i \in \delta(q_{i-1},\varepsilon)$ for any i=1 , . . . , n, in which $q_0 = q$ and $q_n = p$ }.

For any subset X of Q , the ε-closure of X, denoted EC(q), is defined as

$$EC(X) = U_{q\in X}\ EC(q).$$

In particular, EC({q}) = EC(q). Then δ* is defined inductively as

$$\delta^*(q,\varepsilon) = EC(q),$$

$$\delta^*(q,\omega\sigma) = EC(\delta(\delta^*(q,\omega),\sigma)) \text{ for any } q \in Q,\ \omega \in \Sigma^* \text{ and } \sigma \in \Sigma.$$

Then

$$\delta^*(X,\omega) = U_{q\in X}\ \delta^*(q,\omega).$$

It follows that

$$\delta^*(q,\omega\sigma) = \delta^*(\delta^*(q,\omega),\sigma)$$

for any q ∈ Q,ω ∈ Σ* and σ ∈ Σ. By the definition of unitary predicate rec over Σ*, the truth valued rec_A for an l-VFA$_\in$ with crisp transition is defined as follows: for any ω ∈ Σ* ,

$$rec_A(\omega) = V\{F(q) : q \in \delta^*(q_0,\omega)\}.$$

We construct an equivalent l-VAFA β from the above A as follows, where B = (Q, Σ, η, q_0, E). The l-valued transition η is defined as: for any q ∈ Q and σ ∈ Σ,

$\eta(q,\sigma) = \delta^*(q,\sigma)$.

If $q \neq q_0$, then

$E(q) = F(q)$, and $E(q_0) = V\{F(q) : q \in EC(q_0)\}$.

Note that β has no ε-transitions.

III ε-transitions

Lemma 3.1 For any *l*-VAFAε with strong transition A, the I-VFA B constructed as above is equivalent to A,i.e., $rec_A = rec_B$.

Proof. We wish to show by induction on$|\omega|$ that $\eta^*(q,\omega) = \delta^*(q,\omega)$. However, this statement may not be true for $\omega = \varepsilon$, since $\eta^*(q,\omega) = \{q\}$, while $\delta^*(q,\omega) = EC(q)$. We therefore begin our induction at 1.

Let $|\omega| = 1$. Then ω is a symbol σ, and $\eta(q,\sigma) = \delta^*(q,\sigma)$ by definition of η. Suppose that the hypothesis holds for inputs of length n or less. Let $\omega = x\sigma$ be a straight of length of n+1 with symbol σ in Σ. Then

$\eta^*(q, x\sigma) = \eta(\eta^*(q,x),\sigma)$.

By the inductive hypothesis, $\eta^*(q, x) = \delta^*(q, x) = \delta^*(q, x)$. Let $\delta^*(q, x) = X$, we must show that $\eta(X, \sigma) = \delta^*(q, x\sigma)$. But $\eta(X, \sigma) =$

$\mathbf{U}_{q\in Q}\, \eta(q, \sigma) = \mathbf{U}_{q\in X}\, \delta(q,\sigma)$

Then as $X = \delta^*(q, x)$ we have $\mathbf{U}_{q\in Q}\, \delta^*(q, \sigma) = \delta^*(q, x\sigma)$.

Thus $\eta^*(q, x\sigma) = \delta^*(q, x\sigma)$.

To complete the proof we shall show that $rec_B(\omega) = V\{F(q) : q \in \delta^*(q_0,\omega)\}$.

If $\omega = \varepsilon$, this statement is immediate from the definition of E. That is, $\eta^*(q_0,\varepsilon) = \{q_0\}$, then $rec_B(\varepsilon) = V\{E(q) : q \in \eta^*(q_0,\varepsilon) \} = E(q_0) = V\{F(q) : q \in \delta^*(q_0, \varepsilon)\}$.

If $\omega \neq \varepsilon$, then $\omega = x\sigma$ for some symbol σ.We have two cases to discuss.

Case I: $q_0\, \varepsilon'\, \eta^*(q_0,x\sigma)$. by the definition of E and the equality $\eta^*(q_0,x\sigma) = \delta^*(q_0,x\sigma)$, it follows that $rec_{\mathbf{B}}(x\sigma) = V\{E(q) : q \in \eta^*(q_0,x\sigma)\} = V\{F(q) : q \in \delta^*(q_0,x\sigma)\}$.

Case II: $q_0 \in \eta^*(q_0,x\sigma)$. then $EC(q_0) \subseteq \delta^*(q_0,x\sigma)$. Thus,

, $rec_{\mathbf{B}}(x\sigma) = V\{E(q) : q\in \eta^*(q_0,x\sigma)\} = V\{F(q) : q\in \delta^*(q_0,x\sigma)\} =V\{E(q) : q\in\delta^*(q_0,x\sigma) – \{q_0\}\} \vee E(q_0) = V\{F(q) : q \in \delta^*(q_0,x\sigma) – \{q_0\}\} \vee \mathbf{V}\{F(q) : q\in EC(q_0)\} = V\{F(q) : q \in\delta^*(q_0,x\sigma)\}$.

Hence for any $\omega \in \Sigma^*$, $rec_B(\omega) = V\{F(q) : q \in \delta^*(q_0, x\sigma)\} = rec_A(\omega)$. This shows that $rec_A = rec_B$, and thus A and B are equivalent.

Let $A=(Q,\Sigma,\delta,I,F)$ be an I-VFAε. We construct an equivalent I-VFA$_\in$ $B = (P,\Sigma,\eta,S,E)$ with crisp transition from A as follows.

Let $X = Im(I) \cup Im(F)$, and $I_1=X_A$. Choose $P = 2^{Q*(I_1-\{0\})}$, and $S = \{(q,I(q)) : q \in Q \text{ and } I(q) \neq 0\}$,then P is a finite set and $S \in P$. The state transition $\eta : P \times (\Sigma \cup \{\varepsilon\}) \rightarrow P$ is defined by,

$\eta(\{(q,r)\},\tau) = \{(p, r \wedge \delta(q,\tau,p)) : p \in Q \text{ and } r \wedge \delta(q,\tau,p) \neq 0\}$

For any $(q,r) \in Q \times (l_1 - \{0\})$ and $\tau \in \Sigma \cup \{\varepsilon\}$. We define $\eta(Z,\tau) = U_{(q,r) \in Z}\eta(\{(q,r)\},\tau)$

For any $Z \in p$ and $\tau \in \Sigma \cup \{\varepsilon\}$. Then η is well defined as discussed in the extended subset construction from an l-VFA to an l-VDFA. The lattice-valued final state E: P-> l is defined as,

$E(Z) = V\{r \wedge F(q) : (q,r) \in Z\}$.

Lemma 3.2 For any l-VAFA$_\varepsilon$ with crisp transition A the l-VFA B constructed as above is equivalent to A, i.e, $rec_A = rec_B$

Proof: The proof is very similar to that of Theorem 3.1, we omit it here.

Combining the above two lemmas, we can conclude the following theorem which shows the equivalence between l-VAFA$_e$ and l-VAFA.

Theorem 3.1 For any l-VAFA$_e$ A, there is an l-VFA B such that A and B are equivalent, i.e., rec_A ¼ rec_B.

Combining Theorems 3.1 and 3.2, we can see the equivalence between l-VAFA$_e$, l-VAFA and l-VDFA.

Corollary 3.1. For any l-VAFA$_e$ A, there is an l-VDFA B such that A and B are equivalent, i.e., rec_A ¼ rec_B.

As an application of Theorem 3.1, we present pumping lemma in lattice-valued setting as follows, which is independent of the distributive law of the used truth-valued lattice. Qiu has presented the pumping lemma under automata theory based on complete residuated lattice-valued logic recently, and see [26] for the details.

Proposition 3.1 (Pumping lemma in lattice setting). For an l-regular language A : Σ* ? l, there exists positive integer n, for any input string $z \in \Sigma^*$, if $|z| \geq n$, then there are $u,v,w \in \Sigma^*$ such that $|uv| \leq n$, $v \neq \varepsilon$, $z = uvw$, and for any non-negative integer k, the equality $A(uv^kw) = A(uvw)$ holds.

Proof. Since A is l-regular, it is accepted by an l-VDFA $A = (Q, \Sigma,\delta,q_0,F)$ with some particular number of states, say n. Consider an input of n or more symbols $z = \sigma_1 \ldots . \sigma_m$ $m \geq n$, and for i = 1, . . ., m, let $\delta^*(q_0, \sigma_1 \ldots . \sigma_m) = q_i$. It is not possible for each of the n + 1 states q_0, . . ., q_n be different, since there are only n different states. Thus there are two integers j_1 and j_2,

$0 \le j_1 \le j_2 \le n$ such that $q_{j1} = q_{j2}$. Let $u = \sigma_1 \ldots \sigma_{j1}$, $v = \sigma_{j1+1} \ldots \sigma_{j2}$ $w = \sigma_{j2+1} \ldots \sigma_m$, then $|uv| = j_2 \le n$ $v \ne \varepsilon$, and $z = uvw$. Observing that $\delta^*(q_0, \sigma_1 \ldots \sigma_{j1} \sigma_{j2+1} \ldots \sigma_m) = \delta^*(\delta^*(q_0, \sigma_1 \ldots \sigma_{j1}), \sigma_{j2+1} \ldots \sigma_m) = \delta^*(q_{j1}, \sigma_{j2+1} \ldots \sigma_m) = \delta^*(q_{j2}, \sigma_{j2+1} \ldots \sigma_m)$ and for any $k \ge 1$ $\delta^*(q_0, \sigma_1 \ldots \sigma_{j1}(\sigma_{j1+1} \ldots \sigma_{j12})^k \sigma_{j2+1} \ldots \sigma_m) = \delta^*(\delta^*(\delta^*(q_0, \sigma_1 \ldots \sigma_{j1}), (\sigma_{j1+1} \ldots \sigma_{j2})^k), \sigma_{j2+1} \ldots \sigma_m = \delta^*(\delta^*(q_{j1}, \sigma_{j1+1} \ldots \sigma_{j2})^k), \sigma_{j2+1} \ldots \sigma_m = \delta^*(q_{j2}, \sigma_{j2+1} \ldots \sigma_{jm}) = q_m$. Therefore, for any $k \ge 0$, $A(uv^kw)$
$= rec_A(uv^kw) = F(\delta^*(q_0, uv^kw)) = F(q_m) = F(\delta^*(q_0, uvw)) = rec_A(uvw) = A(uvw)$.

4. Kleene Theorem for l-valued finite automata

We use $lR(\Sigma)$ to denote the set of l-regular languages over Σ. In this section, we will study the regular operations on $lR(\Sigma)$, and show that the Kleene theorem (c.f. [12]) holds for l-VFA. As a corollary, we shall show that Kleene theorem holds under quantum logic, which strengthens the results obtained in [33]. Indeed, in [33], the author declared that Kleene theorem in quantum logic was relied on the distributive law of orthomodular lattice, we shall show that this restriction is not necessary.

Let us first give a simple characterization of l-valued regular languages.

Theorem 4.1. Let $A : \Sigma^* \to l$ be an l-valued language over Σ. Then the following statements are equivalent.

(i) A is an l-regular language.

(ii) There exist $k_1, \ldots, k_m \in l \to \_\{0\}$, and regular languages $L_1, \ldots, L_m$ such that $A = V^m_{i=1} k_i\, l_{Li}$, where 1_{Li} denotes the characteristic function of L_i.

(iii) There exist $k_1, \ldots, k_m \in l \to \{0\}$, and pairwise disjoint regular languages $L_1, \ldots, L_m$ satisfying the equality $A = V^m_{i=1} k_i 1_{Li}$.

Proof:

(i) => (iii) Since A is an l-valued regular language, there is an l-VDFA $A = (Q, \Sigma, \delta, q_0, F)$ to recognize A. That is, for all $\omega \in \Sigma^*$ $A(\omega) = rec_A(\omega) = F(\delta^*(q, w))$. Write $Im(F) - \{0\} = \{k_1, \ldots, k_m\}$, and let $F_i = \{q \in Q : F(q) = k_i\}$. For this F_i, we construct a DFA, $A_i = (Q, \Sigma, \delta, q_0, F)$. Let the language recognized by A_i be L_i, then L_i is a regular language, and evidently, the family $\{L_1, \ldots ._m, L_m\}$ is pairwise disjoint. Moreover, $A(\omega) = r$ iff $F(\delta^*(q_0, \omega)) = r$, iff there is i such that $r = k_i$ and $\omega \in L_i$, which shows that $A = V^m_{i=1} k_i\, l_{Li}$

(iii)=>(ii) is obvious

(ii)=>(i) Since each L_i is regular, there is a DFA $A_i = (Q, \Sigma, \delta, q_0, F_i)$ to recognize L_I We can assume that $Q_i = Q_j = \Phi$ whenever $i \ne j$. Define an l=VFA $A = (Q, \Sigma, \delta, q_0, F)$ as follows, $Q = U^m_{i=1} Q_i \cup \{q_0\}$ where $q_0 \notin U^m_{i=1} Q_i$

And $\delta : Q \times \Sigma \to 2^q$ is $\delta(q_0, \sigma) = \{\delta_1 q_{01}, \sigma), \ldots, \delta_m(q_{0m}, \sigma)\}$, for $q \in Q_i, \delta(q, \sigma)\}$; $F(q_0) = \mathbf{V}\{k_i : q_{01} \in F_i\}$ and when $q \ne q_0$,

$$F(q) = \begin{cases} k_i \text{ if } q \in F_i, \\ 0 \text{ otherwise.} \end{cases}$$

Then it can be Then it can be easily verified that $A = recA = V^m_{i=1} k_i\, l_{Li}$. Hence A is an *l*-valued regular language.

We call the *l*-valued language satisfying the condition (ii) or (iii) in the above theorem the *l*-valued recognizable step language, and write the set of all l-valued recognizable languages on

R as step(R), which is equal to lR(Σ).

The following proposition gives the stratified characterization of l-valued recognizable step languages.

Corollary 4.1. Let $A : \Sigma^* \to l$ be an l-valued language over Σ. Then the following statement are equivalent

(i) A is an *l*-regular language.

(ii) The image set Im(A) is finite, and for any $r \in Im(A) - \{0\}$, the r-cut of A, $A_r = \{\omega \in \Sigma^* : A(\omega) \geq r\}$ is a regular language over Σ and $A = \vee_{r \in Im-\{0\}} r 1_{Ar}$.

(iii) The image set Im(A) is finite, and for any $r \in Im(A) - \{0\}$, the r-level of A, $A_{[r]} = \{\omega \in \Sigma^* : A(\omega) = r\}$ is a regular language over Σ and $A = \vee_{r \in Im\_\{0\}} r 1_{A[r]}$.

Theorem 4.2. The family step(Σ) or lR(Σ) is closed under the operations of union, intersection, scalar product, concatenation and Kleene closure.

Proof. Let $A,B \in$ step(Σ). By Theorem 4.1 , we can assume $A = \vee^{m}_{i=1} k_i 1_{Li}$, $B= \vee^{n}_{j=1} k_j 1_{Mj}$, where, all L_i and M_j are regular languages and $\{Li\}^{m}_{i=1}$ are pairwise disjoint, $\{M_j\}^{m}_{i=1}$ are also pairwise disjoint. With respect to the union, we have $A \vee B = \vee^{m}_{i=1} k_i 1_{Li} \vee \vee^{n}_{j=1} k_j 1_{Mj}$ .By Theorem 4.1 , it follows that $A \wedge B \in$ step(Σ).

With respect to the intersection, we have $A \wedge B = \vee^{m}_{i=1} \vee^{n}_{j=1}(k_i \wedge d_{j)}\ 1_{Li \cap Mj}$. By Theorem 4.1 , it follows that $A \wedge B \in$ step(Σ).

With respect to the scalar product, for each $r \in l$, we have $rA(\omega) = r \wedge A(\omega)$, then $rA = \vee^{m}_{i=1}(r \wedge k_i)\ 1_{Li}$ .Therefore, $rA \in$ step(Σ).

For the operation o concatenation, since $AB(\omega) = \vee\{A(\omega_1) \wedge B(\omega_2) : \omega = \omega_1\omega_2\}$, it

For the kleene closure, A* is defined by, $A^*(\omega) = \vee\ \{A(\omega_1) \wedge . . . A(\omega_k) : k \geq 0, \omega = \omega_1 . . . \omega_k\}$ for any $\omega \in \Sigma^*$. Since $A = \vee^{m}_{i=1} k_i 1_{Li}$ and $L_1, . . . , L_m$ are pairwise disjoint regular languages and $k_i \neq 0$ for each i, t follows that $lm(A) – \{0\} = \{k_1, . . . , k_m\}$, and $L_i = \{ \omega \in \Sigma^* : A(\omega) = k_i\}$ (i=1, . . . ,m).For any nonempty subset K of the set {1,2, . . .,m}, ew can assume that $K = \{i_1, . . ., i_s\}$. Let $r_K = k_{i1} \wedge . . . \wedge k_{is}$, $L(K) = \cup_{p1 ... ps} L^{+}_{p1} L^{+}_{p2} L^{*}_{p1} L^{+}_{p3} (L_{p1} \cup L_{p2})^* . . . L^{+}_{ps-1} (L_{p1} \cup . . . \cup L_{ps-2})^* L^{+}_{ps} (L_{p1} \cup . . . \cup L_{ps})^*$, where $p_1 . . . p_s$ is a permutation of $\{i_1, . . ., i_s\}$, and L(K) is taken unions under all permutations of $\{i_1, . . ., i_s\}$. Hence L(K) is a regular language. It is easily verified that $A^* = \vee_{\emptyset \neq K \leq \{1,2,...,m\}} r_k\ 1_{L(K)} \vee 1_{\{\varepsilon\}}$. By Theorem 4.1 , it follows that $A^* \in$ step(Σ).

Remark 4.1. Recall that a negation on a lattice *l* is a mapping h from l into l such that, a 6 b implies h(a) P h(b) and hh(a) = a for any a,b 2 l. Furthermore, if the negation h also satisfies the conditions: a ^ h(a) = 0 and a _ h(a) = 1 for any a 2 l, then h is called the complement over l. In this case, l is an orthocompletment lattice. An orthomodular lattice is an orthocompletment lattice satisfying the orthomodular laws as presented as follows:

$$a \leq b \text{ implies } a \vee (h(a) \wedge b) = b:$$

If *l* is a lattice with a negation h, then for an *l*-language $A : \Sigma^* \to l$, we can define the negation of A on $l(\Sigma^*)$ as, h(A)(x) = h(A(x)). Then for any A ∈ lR(Σ*), h(A) is also l-regular. In fact, if A $=\bigvee_{i=1}^{m}$ h(ki)1_{Σ^*} =1,∨ for some ki ∈ l and regular language Li over Σ,

then h(A) = $=\bigvee_{i=1}^{m}$ h(ki)1_{Σ^*} -1,∨. $l_{\Sigma^*(-L1.....Ln)}$

Definition 4.1 [19] The language of l-valued regular expressions over alphabet ∑ has the alphabet $(\sum\cup\{\varepsilon,\emptyset\})\cup(l\cup\{+,-,*\})$. The symbols in $\sum\cup\{\varepsilon,\emptyset\}$ will be used to denote atomic expressions, and the symbols in $l\cup\{+,-,*\}$ will be used to stand for operators for building up compound expressions: * and all r ∈ l are the unary operators, and +, - are binary ones. We use α, β to act as meta-symbols for regular expressions and L(α) for the language denoted by expression α. More explicitly, L(a) will be used to denote an l-valued subset of ∑*; that is, L(α) ∈ l(∑*). The l-valued regular expressions and the l-valued languages denoted by them are formally defined as follows:

(i) For each σ ∈∑, r is a regular expression, and L(σ) = {σ}; ε and $\emptyset$; are regular expressions, and L(ε) = { ε }, L($\emptyset$) = $\emptyset$.

(ii) If both α and β are regular expressions, then for each r ∈ l, rα, α + β, α - β, α* are all regular expressions, and L(rα) = rL(α), L(α + β) = L(α) ∨ L(β), L(α - β) = L(α)L(β), L(α*) = L(α)*.

Theorem 4.3 (Kleene Theorem in lattice setting). For an l-valued language A ∈ l(∑*), A can be recognized by an l-VFA iff there exists an l-valued regular expression a over ∑ such that A = L(α).

Proof. If A can be recognized by an l-VFA, then by Theorem 4.1, there exist $k_1, \ldots, k_n \in l$ - {0}, and regular languages $L_1, \ldots, L_n$ such that A = $\bigvee_{i=1}^{n} k_i, l_i$. Since each L_i is a regular language, by classical Kleene Theorem, there exists a regular expression α_i over ∑ such that $L(\alpha_i) = L_i$. Let $\alpha = k_1a_1 + \ldots.. + k_na_n$, then α is an l-valued regular expression, and L(α) = $\bigvee_{i=1}^{n} k_i L(\alpha) = \bigvee_{i=1}^{n} k_i l_i$ = A

Conversely, assume that there exists an l-valued regular expression α such that A = L(α). We show that A can be recognized by an l-VFA inductively on the number of operation symbols occurring in α. If there is no operation symbol in α, then α = σ ∈ ∑, ε or $\emptyset$. In this case, L(α) = {σ}, {ε} or $\emptyset$, and L(a) can be recognized by a classical DFA. The classical DFA is evidently an l-VDFA, so L(α) can be recognized by an l-VDFA in this case. Inductively, since the family of recognizable languages by l-VDFA is closed under union, intersection, scalar product, concatenation and Kleene closure (by Theorem 4.2), it follows that L(α) can be recognized by an l-VDFA for any l-valued regular expression α.

Remark 4.2. As a corollary, when l is an orthomodular lattice, Theorem 4.3 is exactly the Kleene theorem in quantum logic. Therefore, the strictly restrictions on Kleene theorem under quantum logic as presented in [33] are not necessary. It also shows that Kleene theorem holds in any lattice-valued finite automata, the distributive law is not necessary to the validity of Kleene theorem.

5.Minimization of lattice-valued finite automata

Finite automata are used to design complex system. Finding a minimum representation of finite automata is a critical is sue arising in such design. In this part, we shall extend the minimal algorithm of deterministic finite automata (DFA) in classical logic [8] to those of lattice setting. To make the meaning of minimization of an *l*-VDFA clear, we first introduce some necessary notions, which are slight modifications of related notions in [20,23].

Given an *l*-VDFA A = Q, $\sum$, δ, q_0, F), let $Q^a = \{d^*(q_0,\theta) : \theta \in \sum^*\}$. If $Q = Q^a$, then A is called accessible. The elements of Q^a are called accessible states, and the elements of Q - Q^a are called inaccessible elements. Consider $\delta^a = \delta|_{Q^a \times \sum^*}$, the restriction mapping of δ on Q^a x $\sum$, and $F^a = F|_{Q^a}$, the restriction mapping on Q^a. Then $A^a = (Q^a, \sum, \delta^a, q_0, F^a)$ is called the accessible part of A. It can be easily shown that these two *l*-VDFAs are equivalent, that is, they accept the same *l*-language, $rec_A = rec_{A^a}$. And we say that an *l*-VDFA is accessible iff $A = A^a$.

Obviously, $\mathcal{A}^a$ is constructed from $\mathcal{A}$ by removing all inaccessible states. Next, we give some notions necessary to compare two or more *l*-VDFAs. Given two *l*-VDFAs $\mathcal{A} = (Q, \sum; \delta; q_0, F)$ and $\mathcal{B} = (P, \sum, \eta, p_0, E)$, a homomorphism from $\mathcal{A}$ to $\mathcal{B}$, denoted $\varphi : \mathcal{A} \to \mathcal{B}$, is a mapping $\varphi : Q \to P$ such that $\varphi(\delta(q,\sigma)) = \eta(\varphi(q),\sigma)$, $\varphi(q_0) = p_0$ and $E(\varphi(q)) \leq F(q)$ for any $q \in Q$ and $\sigma \in \sum$. It can be easily deduced that $\varphi(\delta^*(q,\theta)) = \eta^*(\varphi(q),\theta)$ for any $\theta \in \sum^*$. Furthermore, if $E(p) = \vee\{F(q) : \varphi(q) = p\}$, then φ is called a strong homomorphism. A strong homomorphism $: \mathcal{A} \to \mathcal{B}$ is called an isomorphism if u is one-to-one and onto.

Lemma 5.1. For two *l*-VDFAs $\mathcal{A}$ and $\mathcal{B}$, a homomorphism $\varphi : \mathcal{A} \to \mathcal{B}$ is an isomorphism iff there is another homomorphism $\psi : \mathcal{B} \to \mathcal{A}$ such that $\varphi \circ \psi = 1_{\mathcal{B}}$ and $\psi \circ \varphi = 1_{\mathcal{A}}$.

Proof. ''If part'': In this case, φ is a bijection, i.e., φ is one-to-one and onto. What is left is to show that φ is a strong homomorphism. In fact, for any $q \in Q$, $E(\varphi(q)) \leq F(q) = F(\psi\varphi(q)) \leq E(\varphi(q))$, thus $E(\varphi(q)) = F(q)$''Only if part'': Since φ is an isomorphism, φ is a bijection from Q to P, then there exists an inverse of φ, which is denoted as $\psi : P \to Q$; we show that ψ is also a homomorphism. First, since $\varphi(q_0) = p_0$, $\psi(p_0) = q_0$. Second, if $\psi(p) = q$, then $\varphi(\delta^*(q,\theta)) = \eta^*(\varphi(q), \theta) = \eta^*(p, \theta)$, thus $\psi(\eta^*(p, \theta)) = \delta^*(q, \theta) = \delta^*(\psi(p), \theta)$. Finally, since φ is strong and bijection, $E(p) = F(\psi(p))$.

Lemma 5.2. If $: \mathcal{A}$! B is a homomorphism between two *l*-VDFAs, then $rec_{\mathcal{B}} \leq rec_{\mathcal{A}}$ i.e., $rec_{\mathcal{B}}(\theta) \leq rec_{\mathcal{A}}(\theta)$ for any $\theta \in \sum^*$. Furthermore, if φ is strong, then $rec_{\mathcal{A}} = rec_{\mathcal{B}}$. **Proof.** For any $\theta \in \sum^*$, $rec_{\mathcal{B}}(\theta) = E(\eta^*(\varphi(p_0),\theta))$; $E(\eta^*(\varphi(q_0),\theta)) = E(\varphi(\delta^*(q_0,\theta)) \leq F((\delta^*(q_0,\theta)) = rec_{\mathcal{A}}(\theta)$ that is $rec_{\mathcal{B}} \leq rec_{\mathcal{A}}$. If φ is strong, then $rec_{\mathcal{B}}(\theta) = E(\eta^*(p_0),\theta))$

$= E(\eta^*(\varphi(q_0),\theta)) = E(\varphi(\delta^*(q_0,\theta)) = \vee(F(q): \varphi(q) = \varphi(\delta^*(q_0,\theta) \geq F(\delta^*(q_0,\theta) = rec_{\mathcal{A}}(\theta)$, that is , $rec_{\mathcal{B}} \geq rec_{\mathcal{A}}$, thus $rec_{\mathcal{B}} = rec_{\mathcal{A}}$.

For two equivalent l-VDFAs $\mathcal{A}$ and B, given a homomorphism u : $\mathcal{A} \rightarrow \mathcal{B}$, and consider the restriction of φ on A^a . For any q ∊ Q^a, there exists $\theta \in \sum^*$ such that q = $\delta^*(q_0, \theta)$, in this case, φ (q) = φ ($\delta^*(q_0, \theta)$) = $\eta^*(p_0, \theta)$. Let p = $\eta^*(p_0, \theta)$, then p ∊ P^a. This demonstrates that the restriction of φ on A^a is just the homomorphism $\varphi^a : A^a \rightarrow B^a$. Furthermore, we show that φ^a is also strong and surjective. For any p ∊ P^a, there is $\theta \in \sum^*$ such that p = $\eta^*(p_0, \theta)$. We select q = $\delta^*(q_0, \theta)$. Then q ∊ Q^a and $\varphi^a(q)$ = p, thus φ^a is surjective. Since φ^a is surjective and $F(\delta^*(q_0,\theta) = rec_{\mathcal{A}}(\theta) = rec_{\mathcal{B}}(\theta) = E(\eta^*(q_0),\theta))$ $= E(\varphi(\delta^*(q_0,\theta))$, $\varphi^a$is also a strong homomorphism. We thus obtain the following results.

REFERENCES

[1] 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.

[2] P. Li, Y.M. Li, Algebraic properties of LA-languages, Information Sciences 176 (2006) 3232-3255.

[3] Z.H. Li, P. Li, Y.M. Li, The relationships among several types of fuzzy automata, Information Sciences 176 (2006) 2208-2226.

[4] J.N. Mordeson, D.S. Malik, Fuzzy Automata and Languages: Theory and Applications, Chapman & Hall/CRC, Boca Raton, London, 2002.

[5] D.W. Qiu, Automata theory based on complete residuated lattice-valued logic, Science in China Series F 44 (6) (2001) 419-429.

[6] D.W. Qiu, Automata theory based on complete residuated lattice-valued logic (II), Science in China Series F 45 (6) (2002) 442-452.

[7] D.W. Qiu, Pumping lemma in automata theory based on complete residuated lattice-valued logic, Fuzzy Sets and Systems 157 (2006) 2128-2138.

[8] D.W. Qiu, Automata theory based on quantum logic: reversibilities and pushdown automata, Theoretical Computer Science 386 (2007) 38-56.

[9] D.W. Qiu, Automata theory based on quantum logic: some characterizations, Information and Computation 190 (2007) 179-195.

[10] D.W. Qiu, Notes on automata theory based on quantum logic, Science in China Series F: Information Sciences 50 (2) (2007) 154-169.

[11] M.O. Rabin, D. Scott, Finite automata and their decision problems, IBM Journal of Research and Development 3 (1959) 114-125.

[12] M.P. Schützenberger, On the definition of a family of automata, Information and Control 4 (1961) 245-270.

[13] W. Wechler, The Concept of Fuzzyiness in Automata and Language Theory, Akademie-Verlag, Berlin, 1978.

[14] M.S. Ying, A theory of computation based on quantum logic (I), Theoretical Computer Science 344 (2005) 134-207.