Varieties of fuzzy languages*

Tatjana Petković

Department of Information Technology,
University of Turku, FIN-20014 Turku, Finland
tatpet@utu.fi

Abstract. Fuzzy languages and fuzzy automata are considered using techniques from algebraic theory of automata. Syntactic monoids of fuzzy languages play a crucial role in the study. Varieties of fuzzy languages are introduced and an Eilenberg-type correspondence between varieties of fuzzy languages, varieties of ordinary languages and varieties of monoids is established.

1 Introduction

Fuzzy sets were introduced by Zadeh in [19] and since then have appeared in many fields of sciences. They have been studied within automata theory for the first time by Wee in [18]. More on recent development of algebraic theory of fuzzy automata and formal fuzzy languages can be found in the book by Malik and Mordeson [9]. On the other hand, variety theory establishes correspondences between families of languages, algebras, semigroups and relations. The elementary result of this type is Eilenberg's Variety theorem [4] which was motivated by characterizations of several families of languages by syntactic monoids or semigroups, such as Schützenberger's theorem [14] connecting star-free languages and aperiodic monoids. Eilenberg's theorem has been extended in various directions (see, for example [17, 12, 13, 1, 16, 5, 10, 11]).

This work is an attempt to apply well-known techniques from algebraic theory of ordinary automata in studying fuzzy automata and languages. Syntactic monoids of fuzzy languages are defined and it is shown that they can be computed as transition monoids of recognizers of the fuzzy languages. Moreover, varieties of fuzzy languages are introduced and an Eilenberg-type correspondence between them, varieties of ordinary languages and varieties of finite monoids is established.

Let us recall basic concepts and notation.

A *fuzzy subset* α of a set A is a mapping $\alpha : A \to [0, 1]$. By $\wedge$ and $\vee$ infimum and supremum in the unit segment [0,1] will be denoted, respectively. Clearly, every (ordinary) subset H of A, also called *crisp* subset, can be considered as a

* The work has been supported by the Academy of Finland, decision number 208824. A preliminary version of this work was presented at the Workshop on Semigroups and Automata, a satellite workshop to ICALP'05.

fuzzy subset of A with

$$H(a) = \begin{cases} 1, \, a \in H \\ 0, \, a \notin H \end{cases}$$

Thus when saying $H \subseteq A$ we may mean that H is the above mapping.

A *fuzzy automaton* is a tuple $\mathcal{A} = (A, X, \mu)$, where A is a finite set of *states*, X is a finite set of *input* symbols and μ is a fuzzy subset of $A \times X \times A$ representing the *transition mapping*. This mapping can be represented by the collection of *fuzzy matrices* $\{M(x) = [\mu(a, x, b)]_{a,b \in A} \mid x \in X\}$, i.e., matrices with entries from [0,1].

By X^* the free monoid, i.e., the set of all words with letters from X, is denoted. The empty word is denoted by ε. As proved in [8], mapping μ can be extended to the set $A \times X^* \times A$ by

$$\mu(a, \varepsilon, b) = \begin{cases} 1, \, a = b \\ 0, \, a \neq b \end{cases}$$
$$\mu(a, ux, b) = \textstyle\bigvee_{c \in A}(\mu(a, u, c) \wedge \mu(c, x, b))$$

for all $a, b \in A$, $x \in X$, $u \in X^*$.

A *fuzzy language* over an alphabet X is a fuzzy subset of X^*. A *fuzzy recognizer* is a tuple $\mathcal{A} = (A, X, \mu, \iota, \tau)$ where (A, X, μ) is a fuzzy automaton, ι and τ are fuzzy subsets of A of *initial* and *final* states, respectively. The fuzzy language λ recognized by $\mathcal{A}$ is

$$\lambda(u) = \bigvee_{a \in A} \bigvee_{b \in A} (\iota(a) \wedge \mu(a, u, b) \wedge \tau(b)).$$

A *deterministic fuzzy recognizer* is a fuzzy recognizer with deterministic transition mapping, usually denoted by δ, and crisp one-element set of initial states. It was proved in [2] that a fuzzy language is recognizable by a fuzzy recognizer if and only if it is recognizable by a deterministic fuzzy recognizer.

It can be noticed that fuzzy languages are formal power series and fuzzy recognizers are weighted automata over the semiring $([0, 1], \vee, \wedge, 0, 1)$ (see, for example [6, 3]).

2 Regular fuzzy languages

A fuzzy language is *regular* if it is recognizable by a fuzzy automaton. In this section many concepts and results known for ordinary languages will be extended to fuzzy languages.

Let X be an alphabet, ρ an equivalence on X^* and λ a fuzzy language over X. Then ρ is said to *saturate* λ if $(u, v) \in \rho$ implies $\lambda(u) = \lambda(v)$ for any $u, v \in X^*$. Clearly, in the case of ordinary languages this becomes usual concept of saturation.

Lemma 1. *Let λ be a fuzzy language over an alphabet X. Then the relations defined on X^* by*

$$(u, v) \in P_\lambda \Leftrightarrow (\forall p, q \in X^*) \; \lambda(puq) = \lambda(pvq)$$
$$(u, v) \in R_\lambda \Leftrightarrow (\forall q \in X^*) \; \lambda(uq) = \lambda(vq)$$

are the greatest congruence and the greatest right congruence, respectively, saturating λ.

The monoid X^*/P_λ is the *syntactic monoid* of λ, in notation $\mathrm{Syn}(\lambda)$.

A fuzzy language λ over an alphabet X is *recognizable by a monoid* S if there is a homomorphism $\phi : X^* \to S$ and a fuzzy subset π of S such that $\lambda = \pi\phi^{-1}$, where $\pi\phi^{-1}(u) = \pi(u\phi)$.

A monoid S is said to *divide* a monoid T if S is a homomorphic image of a submonoid of T. The proof of the following theorem resembles proofs of the analogous results for crisp languages.

Theorem 1. (a) *A monoid S recognizes a fuzzy language λ by a homomorphism $\phi : X^* \to S$ if and only if* $\ker\phi$ *saturates λ.*
(b) *A monoid S recognizes a fuzzy language λ if and only if* $\mathrm{Syn}(\lambda)$ *divides S.*

The next theorem represents Myhill-Nerode theorem for fuzzy languages.

Theorem 2. *The following conditions are equivalent for a fuzzy language λ:*

- (i) *λ is regular;*
- (ii) *R_λ has finite index;*
- (iii) *P_λ has finite index;*
- (iv) *λ is recognizable by a finite monoid.*

Proof. Equivalences $(i) \Leftrightarrow (ii) \Leftrightarrow (iii)$ were proved in [9, 15]. Let us only mention here that the recognizer constructed in $(ii) \Rightarrow (i)$ is deterministic recognizer $\mathcal{M}(\lambda) = (A, X, \delta, a_0, \tau)$ where $A = X^*/R_\lambda$, $\delta(u/R_\lambda, x) = (ux)/R_\lambda$, $a_0 = e/R_\lambda$ and $\tau(u/R_\lambda) = \lambda(u)$.

The equivalence $(iii) \Leftrightarrow (iv)$ follows from Theorem 1.

The fuzzy recognizer $\mathcal{M}(\lambda)$ can be constructed in a way similar to the construction of the minimal recognizer for a crisp language. Indeed, the set of states is $\{\lambda.u \mid u \in X^*\}$ where $\lambda.u$ is a fuzzy subset of X^* defined by $\lambda.u(w) = \lambda(uw)$ for any $w \in X^*$. It can be easily seen that $\lambda.u = \lambda.v$ if and only if $u/R_\lambda = v/R_\lambda$. Then transitions are defined by $\delta(\lambda.u, x) = \lambda.(ux)$, $\lambda = \lambda.\varepsilon$ is the initial state, and $\tau(\lambda.u) = \lambda(u)$.

The *transition monoid* $T(\mathcal{A})$ of a fuzzy automaton $\mathcal{A} = (A, X, \mu)$ can be defined in two equivalent ways. Namely, let $\theta_\mathcal{A}$ be the relation defined on X^* by

$$(u, v) \in \theta_\mathcal{A} \Leftrightarrow (\forall a, b \in A)\ \mu(a, u, b) = \mu(a, v, b).$$

Then $\theta_\mathcal{A}$ is a congruence and $T(\mathcal{A}) = X^*/\theta_\mathcal{A}$, see [7]. On the other hand, $T(\mathcal{A}) = \{M(u) \mid u \in X^*\}$ is a monoid consisting of fuzzy matrices of type $A \times A$, with multiplication $\circ$ defined by

$$(M(u) \circ M(v))_{ab} = \bigvee_{c \in A} (M(u)_{ac} \wedge M(v)_{cb}),\ a, b \in A.$$

It is generated by the set $\{M(x) \mid x \in X\}$. In the case of deterministic fuzzy automata, i.e., ordinary automata, the relation $\theta_{\mathcal{A}}$ becomes Myhill congruence and $T(\mathcal{A})$ is the usual transition monoid of $\mathcal{A}$.

Using the following theorem the monoid $\mathrm{Syn}(\lambda)$, for a fuzzy language λ, can be computed.

Theorem 3. *For a fuzzy language λ, $T(\mathcal{M}(\lambda)) \cong \mathrm{Syn}(\lambda)$.*

Proof. Follows directly from the fact $P_\lambda = \theta_{\mathcal{M}(\lambda)}$.

It is well-known that not every monoid is a syntactic monoid of a crisp language. This is not the case with fuzzy languages.

Theorem 4. *Every monoid is the syntactic monoid of a fuzzy language.*

Proof. Consider a monoid S and an alphabet X such that there is an epimorphism $\phi : X^* \to S$. Let $\{L_i\}_{i\in I}$ be languages in the partition determined by $\ker \phi$. Let $\{c_i\}_{i\in I}$ be pairwise distinct numbers from [0,1]. The fuzzy language $\lambda : X^* \to [0,1]$ is defined by $\lambda(u) = c_i \Leftrightarrow u \in L_i$. It can be proved that the mapping $\psi : \mathrm{Syn}(\lambda) \to S$ defined by $(u/P_\lambda)\psi = u\phi$ is an isomorphism.

3 Varieties of fuzzy languages

Recall that a family $\mathscr{C} = \{\mathscr{C}(X)\}$ of regular (crisp) languages is a *variety of languages* if it is closed under Boolean operations, quotients and inverse homomorphic images. These are exactly families of crisp languages definable by varieties of monoids. We are going to describe here the corresponding families of fuzzy languages.

For fuzzy languages $\lambda, \lambda_1, \lambda_2$ over an alphabet X, *complement, union* and *intersection* are defined respectively by

$$\begin{aligned}&\overline{\lambda}(u) = 1 - \lambda(u)\\&(\lambda_1 \vee \lambda_2)(u) = \lambda_1(u) \vee \lambda_2(u)\\&(\lambda_1 \wedge \lambda_2)(u) = \lambda_1(u) \wedge \lambda_2(u).\end{aligned}$$

The set of all fuzzy languages over the same set together with these operations forms a De Morgan algebra.

Further, *left* and *right quotients* are defined respectively by

$$\begin{aligned}&(\lambda_1^{-1}\lambda_2)(u) = \textstyle\bigvee_{v\in X^*}(\lambda_2(vu) \wedge \lambda_1(v))\\&(\lambda_2\lambda_1^{-1})(u) = \textstyle\bigvee_{v\in X^*}(\lambda_2(uv) \wedge \lambda_1(v)).\end{aligned}$$

Let $c \in [0,1]$ be arbitrary, then the fuzzy language $c\lambda$ is defined by

$$(c\lambda)(u) = c \cdot \lambda(u).$$

Let $\phi : X^* \to Y^*$ be a homomorphism and τ a fuzzy language over Y, then the *inverse image* of τ is a fuzzy language $\tau\phi^{-1}$ over X defined by

$$(\tau\phi^{-1})(u) = \tau(u\phi).$$

Lemma 2. *Let X and Y be alphabets, $\lambda, \lambda_1, \lambda_2$ be fuzzy languages over X, τ is a fuzzy language over Y, $\phi : X^* \to Y^*$ a homomorphism and $c \in (0, 1]$ arbitrary constant. Then the following holds:*

(a) $P_{\overline{\lambda}} = P_{c\lambda} = P_\lambda$
$P_{\lambda_1 \vee \lambda_2}, P_{\lambda_1 \wedge \lambda_2} \supseteq P_{\lambda_1} \cap P_{\lambda_2}$;
(b) $P_{\lambda_1^{-1}\lambda_2}, P_{\lambda_2\lambda_1^{-1}} \supseteq P_{\lambda_2}$;
(c) $\phi \circ P_\tau \circ \phi^{-1} \subseteq P_{\tau\phi^{-1}}$, *where* $\phi \circ P_\tau \circ \phi^{-1}$ *is a congruence on* X^* *defined by* $(u, v) \in \phi \circ P_\tau \circ \phi^{-1} \Leftrightarrow (u\phi, v\phi) \in P_\tau$.

As a consequence of Lemma 2 we get the following result.

Corollary 1. *Complements, unions, intersections, products by constants, quotients and inverse homomorphic images of regular fuzzy languages are regular.*

For a fuzzy language λ by a *c-cut*, $c \in [0, 1]$, we mean the crisp language λ_c defined by

$$\lambda_c = \{u \in X^* \mid \lambda(u) \geq c\}.$$

It is easy to see that

$$\lambda = \bigvee_{c \in [0,1]} c\lambda_c.$$

The following theorem gives a very useful connection between regular fuzzy and crisp languages.

Theorem 5 ([15, 9]). *For a fuzzy language λ, $P_\lambda = \bigcap_{c\in[0,1]} P_{\lambda_c}$.*

Moreover, a fuzzy language λ is regular if and only if $\mathrm{Im}(\lambda)$ *is finite and language λ_c is regular for every $c \in [0, 1]$.*

A family $\mathscr{F} = \{\mathscr{F}(X)\}$ of regular fuzzy languages is a *variety of fuzzy languages* if it is closed under unions, intersections, complements, multiplications by constants, quotients, inverse homomorphic images and cuts.

Let $\mathscr{F}$ and $\mathscr{C}$ be varieties of fuzzy and crisp languages, respectively, and let us define the assignments $\mathscr{F} \mapsto \mathscr{F}^c$ and $\mathscr{C} \to \mathscr{C}^f$ by

$$\mathscr{F}^c(X) = \{L \subseteq X^* \mid L \in \mathscr{F}(X) \text{ is crisp}\}$$
$$\mathscr{C}^f(X) = \{\lambda \text{ is a fuzzy language over } X \mid \lambda = \textstyle\bigvee_{i=1}^n c_i L_i \text{ for some } n \in \mathbb{N}, c_i \in [0, 1], L_i \in \mathscr{C}(X)\}.$$

Lemma 3. *Let $\mathscr{F}, \mathscr{F}_1, \mathscr{F}_2$ be varieties of fuzzy languages and let $\mathscr{C}, \mathscr{C}_1, \mathscr{C}_2$ be varieties of crisp languages. Then:*

(a) $\mathscr{F}^c$ *is a variety of crisp languages;*
(b) $\mathscr{C}^f$ *is a variety of fuzzy languages;*
(c) $\mathscr{F}_1(X) \subseteq \mathscr{F}_2(X)$ *implies* $\mathscr{F}_1^c(X) \subseteq \mathscr{F}_2^c(X)$ *for every* X;
(d) $\mathscr{C}_1(X) \subseteq \mathscr{C}_2(X)$ *implies* $\mathscr{C}_1^f(X) \subseteq \mathscr{C}_2^f(X)$ *for every* X;
(e) $\mathscr{F}^{cf} = \mathscr{F}$;
(f) $\mathscr{C}^{fc} = \mathscr{C}$.

Proof. The claim (a) follows from the fact that concepts of Boolean operations, quotients and inverse images extend the usual ones defined for crisp languages. Proofs for (c) and (d) are straightforward. Claims (e) and (f) are also not difficult to prove.

Finally, let us prove (b). Boolean closure of the family $\mathscr{C}^f(X)$ follows from the following identities

$$\begin{aligned}&\overline{cL} = (1-c)L \vee \overline{L}\\ &c_1L_1 \wedge c_2L_2 = (c_1 \wedge c_2)(L_1 \cap L_2)\end{aligned}$$

and the fact that $\mathscr{C}(X)$ is closed under Boolean operations. The family is obviously closed under products by constants. Considering cuts, it can be proved that for $\lambda = \bigvee_{i=1}^n c_iL_i$ it follows that $\lambda_c = \bigcup_{c_i \geq c} L_i$, and so $\lambda_c \in \mathscr{C}(X) \subseteq \mathscr{C}^f(X)$. Let now $\phi : X^* \to Y^*$ be a homomorphism and $\lambda \in \mathscr{C}^f(Y)$. Then $\lambda = \bigvee_{i=1}^n c_iL_i$ for some $L_i \in \mathscr{C}(Y), c_i \in [0,1], n \in \mathbb{N}$. Then $\lambda\phi^{-1}(u) = \lambda(u\phi) = \bigvee_{i=1}^n c_iL_i(u\phi) = \bigvee_{i=1}^n c_i(L_i\phi^{-1})(u)$. Since $L_i\phi^{-1} \in \mathscr{C}(X)$, then $\lambda\phi^{-1} \in \mathscr{C}^f(X)$.

Let us prove now that $\mathscr{C}^f(X)$ is closed for quotients. Let $\lambda = \bigvee_{i=1}^n c_iL_i \in \mathscr{C}^f(X)$ and τ be a fuzzy language over X. Then

$$\begin{aligned}\tau^{-1}\lambda(u) &= \bigvee_{v\in X^*}(\lambda(vu) \wedge \tau(v))\\ &= \bigvee_{v\in X^*}(\bigvee_{i=1}^n c_iL_i(vu) \wedge \tau(v))\\ &= \bigvee_{v\in X^*}\bigvee_{i=1}^n(c_iL_i(vu) \wedge \tau(v))\\ &= \bigvee_{i=1}^n\bigvee_{v\in X^*}(c_iL_i(vu) \wedge \tau(v))\\ &= \bigvee_{i=1}^n\bigvee_{v\in X^*}(c_i(v^{-1}L_i)(u) \wedge \tau(v))\end{aligned}$$

The fact that L_i is regular implies that the principal right congruence R_{L_i} has finite index. Let $A_{i_1}, A_{i_2}, \ldots, A_{i_k}$, $k \in \mathbb{N}$, be the partition of X^* determined by R_{L_i}. Clearly, the value of $c_i(v^{-1}L_i)(u)$ does not depend on the choice of $v \in A_{i_j}$, it depends only on i_j, $j \in \{1,\ldots,k\}$. Let us also denote $t_{i_j} = \bigvee_{v\in A_{i_j}} \tau(v)$. Then we can continue the previous computation as follows

$$\begin{aligned}&= \bigvee_{i=1}^n\bigvee_{j=1}^k\bigvee_{v\in A_{i_j}}(c_i(v^{-1}L_i)(u) \wedge \tau(v))\\ &= \bigvee_{i=1}^n\bigvee_{j=1}^k(c_i(v^{-1}L_i)(u) \wedge \bigvee_{v\in A_{i_j}}\tau(v))\\ &= \bigvee_{i=1}^n\bigvee_{j=1}^k(c_i(v^{-1}L_i)(u) \wedge t_{i_j})\\ &= \bigvee_{i=1}^n\bigvee_{j=1}^k(c_i \wedge t_{i_j})(v^{-1}L_i)(u)\end{aligned}$$

Now from $v^{-1}L_i \in \mathscr{C}(X)$ follows that $\tau^{-1}\lambda \in \mathscr{C}^f(X)$. Closure for right quotients can be proved similarly.

Now we can formulate the following theorem, whose proof follows directly from Lemma 3.

Theorem 6. *The mappings $\mathscr{F} \mapsto \mathscr{F}^c$ and $\mathscr{C} \mapsto \mathscr{C}^f$ are mutually inverse lattice isomorphisms between the lattices of all varieties of fuzzy languages and all varieties of crisp languages.*

4 Variety theorem

Recall that a set $\mathscr{S}$ of finite monoids is a *variety of finite monoids* if it is closed under taking submonoids, homomorphic images and finite direct products. To a variety of crisp languages $\mathscr{C}$ a variety of finite monoids $\mathscr{C}^s$ generated by syntactic monoids of languages from $\mathscr{C}$ is assigned. On the other hand, to a variety of finite monoids $\mathscr{S}$ a variety $\mathscr{S}^c$ of regular languages with syntactic monoids in $\mathscr{S}$ is assigned. According to the famous Eilenberg's theorem, the mappings $\mathscr{C} \mapsto \mathscr{C}^s$ and $\mathscr{S} \to \mathscr{S}^c$ are mutually inverse lattice isomorphisms between the lattices of all varieties of languages and all varieties of finite monoids.

Eilenberg's theorem and Theorem 6 imply that the lattices of all varieties of fuzzy languages and all varieties of finite monoids are isomorphic. In this section we are going to establish mutual isomorphisms, and their connection with isomorphisms used in Eilenberg's theorem and Theorem 6.

For a variety of fuzzy languages $\mathscr{F}$, let $\mathscr{F}^s$ be the family of finite monoids defined by

$$\mathscr{F}^s = \{\mathrm{Syn}(\lambda) \mid \lambda \in \mathscr{F}(X) \text{ for some } X\}.$$

It can be proved that $\mathscr{F}^s$ is a variety of monoids using arguments similar to the proofs of Theorem 4 and Proposition 3.10 [1].

On the other hand, for a variety of finite monoids $\mathscr{S}$, let $\mathscr{S}^f = \{\mathscr{S}^f(X)\}$ be the family of fuzzy languages defined by

$$\mathscr{S}^f(X) = \{\lambda \text{ is a fuzzy language over X} \mid \mathrm{Syn}(\lambda) \in \mathscr{S}\}.$$

The fact that $\mathscr{S}^f$ is a variety of fuzzy languages follows from Lemma 2 and the first part of Theorem 5.

Using this terminology, according to Eilenberg's theorem, $\mathscr{C}^{sc} = \mathscr{C}$ and $\mathscr{S}^{cs} = \mathscr{S}$ hold. More relationships of this kind between the introduced operators are proved in the following lemma.

Lemma 4. *Let $\mathscr{F}$, $\mathscr{C}$ and $\mathscr{S}$ be, respectively, a variety of fuzzy languages, variety of crisp languages and variety of finite monoids. Then:*

(a) $\mathscr{F}^{cs} = \mathscr{F}^s$ *and* $\mathscr{C}^{fs} = \mathscr{C}^s$*;*
(b) $\mathscr{S}^{fc} = \mathscr{S}^c$*;*
(c) $\mathscr{S}^{cf} = \mathscr{S}^f$*;*
(d) $\mathscr{F}^{sc} = \mathscr{F}^c$ *and* $\mathscr{C}^{sf} = \mathscr{C}^f$*.*

Proof. (*a*) The inclusion $\mathscr{F}^{cs} \subseteq \mathscr{F}^s$ is obvious. For the opposite inclusion, it suffices to prove that $\mathrm{Syn}(\lambda) \in \mathscr{F}^{cs}$ for every $\lambda \in \mathscr{F}(X)$. Indeed, by definition of varieties of fuzzy languages, it follows that $\lambda_c \in \mathscr{F}(X)$, and thus $\lambda_c \in \mathscr{F}^c(X)$, for every $c \in [0,1]$. According to Theorem 5, it follows that $\mathrm{Syn}(\lambda)$ is a subdirect product of $\mathrm{Syn}(\lambda_c)$, $c \in [0,1]$, and so $\mathrm{Syn}(\lambda) \in \mathscr{F}^{cs}$.

Using Theorem 6 and the already proved equality $\mathscr{F}^s = \mathscr{F}^{cs}$, we get $\mathscr{C}^s = \mathscr{C}^{fcs} = \mathscr{C}^{fs}$.

(*b*) Clearly, $L \in \mathscr{S}^{fc}(X)$ if and only if $L \in \mathscr{S}^f(X)$ is a crisp language, i.e., if and only if $\mathrm{Syn}(L) \in \mathscr{S}$ for a crisp language L over an alphabet X, and this holds if and only if $L \in \mathscr{S}^c(X)$.

(c) According to Theorem 6 and (b), we have $\mathscr{S}^f = \mathscr{S}^{fcf} = \mathscr{S}^{cf}$.
(d) Applying Theorem 6, (a) and Eilenberg's theorem, we get $\mathscr{F}^{sc} = \mathscr{F}^{cfsc} = \mathscr{F}^{csc} = \mathscr{F}^c$. Similarly, using (c) and Eilenberg's correspondence, we get $\mathscr{C}^{sf} = \mathscr{C}^{scf} = \mathscr{C}^f$.

The following theorem is a counterpart of Eilenberg's theorem for varieties of fuzzy languages.

Theorem 7. *The mappings $\mathscr{F} \mapsto \mathscr{F}^s$ and $\mathscr{S} \mapsto \mathscr{S}^f$ are mutually inverse lattice isomorphisms between the lattices of all varieties of fuzzy languages and all varieties of finite monoids.*

Proof. Let $\mathscr{F}, \mathscr{F}_1, \mathscr{F}_2$ be varieties of fuzzy languages and $\mathscr{S}, \mathscr{S}_1, \mathscr{S}_2$ varieties of finite monoids. It is easy to check that $\mathscr{S}_1 \subseteq \mathscr{S}_2$ implies $\mathscr{S}_1^f(X) \subseteq \mathscr{S}_2^f(X)$ for every X, and $\mathscr{F}_1(X) \subseteq \mathscr{F}_2(X)$ for every X implies $\mathscr{F}_1^s \subseteq \mathscr{F}_2^s$. According to Eilenberg's theorem $\mathscr{S} = \mathscr{S}^{cs}$, and Lemma 4, we have $\mathscr{S} = \mathscr{S}^{cs} = \mathscr{S}^{fcs} = \mathscr{S}^{fs}$. Finally, using Theorem 6, Eilenberg's theorem and Lemma 4, we get $\mathscr{F} = \mathscr{F}^{cf} = \mathscr{F}^{cscf} = \mathscr{F}^{csf} = \mathscr{F}^{sf}$, what finishes the proof.

Example 1. Let us consider the family $\mathrm{Hom}(X)$ of all fuzzy languages over an alphabet X that are homomorphisms from X^* to $([0,1], \wedge)$, i.e., $\lambda \in \mathrm{Hom}(X)$ if and only if $\lambda(uv) = \lambda(u) \wedge \lambda(v)$ for any $u, v \in X^*$. The family $\mathrm{Hom}(X)$ is closed under taking intersections, products by constants, cuts, but not for unions and complements. Let $\mathscr{F} = \{\mathscr{F}(X)\}$ be the smallest variety containing $\mathrm{Hom}(X)$. Then $\mathscr{F}^s$ is the variety of finite semilattices, whereas the corresponding variety of crisp languages is $\mathscr{F}^c = \{\mathscr{F}^c(X)\}$ with $\mathscr{F}^c(X)$ the Boolean closure of the family of crisp languages $\{Y^* \mid Y \subseteq X\}$ for any alphabet X.

5 Conclusion and further work

Standard algebraic techniques known from algebraic theory of automata and formal languages are used in studying fuzzy automata and fuzzy languages. Regular fuzzy languages are considered in terms of recognition by finite monoids. A connection with recognition by fuzzy automata is given. Varieties of fuzzy languages are defined so that they are exactly families of regular fuzzy languages determined by varieties of finite monoids. Thus, an Eilenberg-type correspondence between varieties of fuzzy languages, varieties of crisp languages and varieties of finite monoids is proved. Knowing that fuzzy languages are a special kind of formal power series, it remains for future research to generalize the correspondences established here to formal power series over more general semirings.

References

1. J. Almeida, On pseudovarieties, varieties of languages, filters of congruences, pseudoidentities and related topics, *Algebra Universalis* **27** (1990), 333–350.

2. R. Bělohlávek, Determinism and fuzzy automata, *Information Sciences* **143** (2002) 205–209.
3. M. Droste and G.-Q. Zhang, On transformations of formal power series, *Information and Computation* **184** (2003), 369–383.
4. S. Eilenberg, *Automata, Languages, and Machines*, Vol. **B.** Pure and Applied Mathematics, Vol. 59, Academic Press, New York – London (1976).
5. Z. Ésik, A variety theorem for trees and theories, in: *Automata and formal languages* VIII (Salgótarján, 1996), *Publ. Math. Debrecen* **54** (1999), 711–762.
6. W. Kuich, Semirings and formal power series, in: *Handbook of Formal Languages*, (Eds.: G. Rozenberg and A. Salomaa), Springer, 1997, Vol. 1, Chapter 9, 609–677.
7. D. S. Malik, J. N. Mordeson and M. K. Sen, Products of fuzzy finite state machines, *Fuzzy Sets and Systems* **92** (1997) 95–102.
8. D. S. Malik, J. N. Mordeson and M. K. Sen, Minimization of fuzzy finite automata, *Information Sciences* **113** (1999) 323–330.
9. J. N. Mordeson and D. S. Malik, *Fuzzy automata and languages*, Chapman & Hall/CRC, 2002.
10. T. Petković, M. Ćirić and S. Bogdanović, Unary algebras, semigroups and congruences on free semigroups, *Theoretical Computer Science* **324:1** (2004), 87-105.
11. T. Petković and S. Salehi, Positive Varieties of Tree Languages, *Theoretical Computer Science* (to appear), available as: TUCS Technical Report 622, September 2004.
URL: `http://www.tucs.fi/publications/insight.php?id=tSaPe04a`
12. J. E. Pin, *Varieties of formal languages*, Foundations of Computer Science, Plenum Publishing Corp., New York, 1986.
13. J. E. Pin, A variety theorem without complementation, *Izvestiya VUZ Matematika* **39** (1995), 80–90. English version, *Russian Mathem.* (*Iz. VUZ*) **39** (1995), 74–63.
14. M. P. Schützenberger, On finite monoids having only trivial subgroups, *Information and Control* **8** (1965), 190–194.
15. J. Shen, Fuzzy language on free monoid, *Information Sciences* **88** (1996), 149–168.
16. M. Steinby, General varieties of tree languages, *Theoretical Computer Science* **205** (1998), 1–43.
17. D. Thérien, Recognizable languages and congruences, *Semigroup Forum* **23** (1981), 371–373.
18. W. G. Wee, *On generalizations of adaptive algorithm and application of the fuzzy sets concept to pattern classification*, Ph.D. Thesis, Purdue University, 1967.
19. L. A. Zadeh, Fuzzy sets, *Information and Control* **8** (1965) 338–353.