View metadata, citation and similar papers at core.ac.uk brought to you by CORE

FACTA UNIVERSITATIS (NIŠ)
Ser. Math. Inform. Vol. 30, No 3 (2015), 235–253

DIFFERENT MODELS OF AUTOMATA WITH FUZZY STATES*

Aleksandar Stamenković, Miroslav Ćirić, and Jelena Ignjatović

Abstract. In this paper we provide a general definition of automata with fuzzy states which includes as its special cases automata used by Lin et al. [29], Liu and Qiu [30,31,42] and Xing et al. [56] in the study of fuzzy discrete event systems, as well as various types of automata constructed in [14, 15, 18, 32] for the purpose of determinization of fuzzy automata. We explain the relationships between these different models of automata with fuzzy states and show that every crisp-deterministic fuzzy automaton can be transformed into a language-equivalent automaton with fuzzy states, and vice versa.
Keywords: Fuzzy automaton, fuzzy language, crisp-deterministic fuzzy automaton, automaton with fuzzy states, Nerode automaton, derivative automaton, prefix-closure, fuzzy relation equation.

1. Introduction

From the very beginning of the theory of fuzzy sets, fuzzy automata and languages are studied as a means for bridging the gap between the precision of computer languages and vagueness and imprecision, which are frequently encountered in the study of natural languages. The study of fuzzy automata and languages was initiated in 1960s by Santos [43–45], Wee [49], Wee and Fu [50], and Lee and Zadeh [22]. From late 1960s until early 2000s mainly fuzzy automata and languages with membership values in the Gödel structure have been considered (cf., e.g., [11, 12, 33]). The idea of studying fuzzy automata with membership values in some structured abstract set comes back to Wechler [48], and in recent years researchers' attention has been aimed mostly at fuzzy automata with membership values in complete residuated lattices, lattice-ordered monoids, and other kinds of lattices. Fuzzy automata taking membership values in a complete residuated lattice were first studied by Qiu in [38, 39], where some basic concepts were discussed, and later, Qiu and his coworkers carried out extensive research into these fuzzy automata (cf. [40, 41, 51–55]). From a different point of view, fuzzy

Received February 04, 2015; Accepted April 11, 2015
2010 *Mathematics Subject Classification.* Primary 68Q45, 68Q70, 03E72
*Research supported by Ministry of Education, Science and Technological Development, Republic of Serbia, Grant No. 174013

automata with membership values in a complete residuated lattice were studied by Ignjatović, Ćirić and their coworkers in [5–8, 13–18, 32, 46, 47]. Fuzzy automata taking membership values in a lattice-ordered monoid were investigated by Li and others [23, 24, 26, 28], fuzzy automata over other types of lattices were the subject of [2, 10, 20, 21, 25, 27, 34–37], and automata which generalize fuzzy automata over any type of lattices, as well as weighted automata over semirings, were studied recently in [4, 9, 17]. For decades, fuzzy automata and languages have gained a wide field of application, including lexical analysis, description of natural and programming languages, learning systems, control systems, neural networks, knowledge representation, clinical monitoring, pattern recognition, error correction, databases, discrete event systems, and many other areas (cf., e.g., [11, 12, 19, 33, 36]).

Fuzzy automata are a natural generalization of ordinary nondeterministic automata, and the most natural generalization of ordinary deterministic automata are those fuzzy automata which have a single crisp initial state and a deterministic transition function, and the fizziness is entirely concentrated in the fuzzy set of terminal states. These automata were first introduced by Bělohlávek [2], and in [4] they were called crisp-deterministic fuzzy automata. Determinization of fuzzy automata, i.e., the problem of their conversion into language-equivalent crisp-deterministic fuzzy automata, was also first studied by Bělohlávek [2], in the context of fuzzy finite automata over a complete distributive lattice, and Li and Pedrycz [26], in the context of fuzzy finite automata over a lattice-ordered monoid. Determinization algorithms that were provided there generalize the subset construction. Another algorithm, provided by Ignjatović et al. [14], also generalizes the subset construction and produces a smaller crisp-deterministic fuzzy automaton than algorithms from [2, 26], regardless of the fuzzy automaton which is determinized. This crisp-deterministic fuzzy automaton can be alternatively constructed by means of the Nerode right congruence of the original fuzzy finite automaton, and it was called in [15] the Nerode automaton of the original fuzzy finite automaton. The Nerode automaton was constructed in [14] for fuzzy finite automata over a complete residuated lattice, and it was noted that the identical construction can also be made in a more general context, for fuzzy finite automata over a lattice-ordered monoid, and even for weighted finite automata over a semiring. The same construction was also transferred in [4] to weighted automata over strong bimonoids. The algorithm proposed by Jančić et al. in [17] produces a crisp-deterministic fuzzy or weighted automaton that is even smaller than the Nerode automaton, and further progress has been made in a recent paper by Jančić et al. [18], where algorithms which perform both determinization and state reduction have been provided. In addition, Jančić and Ćirić in [16] adapted the well-known Brzozowski's double reversal determinization algorithm to fuzzy automata and provided a Brzozowski type determinization algorithm that yields a minimal crisp-deterministic fuzzy automaton equivalent to the original fuzzy finite automaton. Another algorithm which constructs a minimal crisp-deterministic fuzzy automaton equivalent to the original fuzzy finite automaton, which is theoretically faster than the Brzozowski type algorithm, has been provided by Micić et al. in [32].

Another model of fuzzy automata, called here automata with fuzzy states, has been discussed in a series of papers dealing with fuzzy discrete event systems (cf. [29–31, 42, 56]). The set of states of such an automaton is the collection of all fuzzy subsets of a given set A, the initial state is a single fuzzy subset of A, and the set of terminal states (or marking states, as they were called in the listed papers) is any collection of fuzzy subsets of A. The transitions are defined by means of compositions of fuzzy states with pre-specified fuzzy relations which were called fuzzy events. We say that such transitions are compositionally defined. Alternatively, the set of fuzzy events can be understood as a family of fuzzy transition relations indexed by an ordinary crisp set of events (or inputs). Here we give a somewhat different definition of an automaton with fuzzy states. To provide a definition of these automata which includes some important automata, we assume that the set of fuzzy states is any collection (possibly finite) of fuzzy subsets of A. The initial state is also a single fuzzy subset of A, and "terminal states" are modeled by a fuzzy subset of the set of fuzzy states. Finally, transitions are defined by means of a function which maps the Cartesian product of the set of fuzzy states and the input alphabet into the set of fuzzy states.

The main aim of this paper is to compare automata defined in such a way with some related types of automata. By Theorem 3.1 we show that any crisp-deterministic fuzzy automaton can be transformed into a language-equivalent automaton with fuzzy states. The reverse transformation is simple because we have only to ignore the fuzzy nature of the states. Then we show that the Nerode automaton of a fuzzy automaton $\mathcal{A}$ can be viewed as an automaton with fuzzy states, and by Theorem 3.2 we prove that such Nerode automaton is completely language-equivalent to $\mathcal{A}$. Theorem 3.3 provides necessary and sufficient conditions under which an automaton with fuzzy states can be compositionally defined. These conditions are stated in terms of solvability of particular systems of fuzzy relation equations. Using this result, by Example 3.1 we demonstrate that there are automata with fuzzy states which can not be compositionally defined, which means that our definition is more general then the one used in [29–31, 42, 56]. We show that other types of crisp-deterministic fuzzy automata used in [18, 32] in the determinization of fuzzy automata can also be considered as automata with fuzzy states.

Yet another type of automata that can be treated as automata with fuzzy states are the so-called derivative automata introduced in [15]. The derivative automaton one constructs starting from a given fuzzy language f and its derivatives, and it has been proved in [15] that it is a minimal crisp-deterministic fuzzy automaton which recognizes f. Here we prove that the derivative automaton of f is also a minimal automaton with fuzzy states which recognizes f and generates the prefix-closure of f (Theorem 3.4), and as a consequence we obtain that every prefix-closed fuzzy language can be generated by an automaton with fuzzy states. Previously, by Theorem 2.1 we have proved that any fuzzy language generated by a fuzzy automaton is prefix-closed. We also provide an example of a fuzzy language generated by an automaton with fuzzy states which is not-prefix closed. Finally, by Theorem 3.5 we prove solvability of some other systems of fuzzy relation equations, which implies that the derivative automaton of an arbitrary fuzzy language can also be compositionally defined.

The paper has three sections. After this introductory section, in Section 2 we give the basic notions and notation concerning fuzzy sets and relations and fuzzy automata and languages. The main results of the paper, which have been mentioned above, are presented in Section 3.

2. Preliminaries

In this section we introduce the basic notions and notation concerning fuzzy sets and relations and fuzzy automata and languages.

2.1. Fuzzy sets and relations

In this paper we use complete residuated lattices as structures of membership values. A *residuated lattice* is an algebra $\mathcal{L} = (L, \wedge, \vee, \otimes, \to, 0, 1)$ 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 $\to$ form an *adjoint pair*, i.e., they satisfy the *adjunction property*: for all $x, y, z \in L$,

$$x \otimes y \leqslant z \iff x \leqslant y \to z. \tag{2.1}$$

If, additionally, $(L, \wedge, \vee, 0, 1)$ is a complete lattice, then $\mathcal{L}$ is called a *complete residuated lattice*.

The operations $\otimes$ (called *multiplication*) and $\to$ (called *residuum*) are intended for modeling the conjunction and implication of the corresponding logical calculus, and supremum ($\bigvee$) and infimum ($\bigwedge$) are intended for modeling of the existential and general quantifier, respectively. An operation $\leftrightarrow$ defined by

$$x \leftrightarrow y = (x \to y) \wedge (y \to x), \tag{2.2}$$

called *biresiduum* (or *biimplication*), is used for modeling the equivalence of truth values. It can be easily shown that with respect to $\leqslant$, $\otimes$ is isotonic in both arguments, $\to$ is isotonic in the second and antitonic in the first argument, and for any $x, y, z \in L$ the following hold:

$$(x \to y) \otimes x \leqslant y, \tag{2.3}$$

$$y \leqslant x \to (x \otimes y), \tag{2.4}$$

For other properties of complete residuated lattices one can refer to [1,3].

The most studied and applied structures of truth values, defined on the real unit interval $[0, 1]$ with $x \wedge y = \min(x, y)$ and $x \vee y = \max(x, y)$, are the *Łukasiewicz structure* ($x \otimes y = \max(x + y - 1, 0)$, $x \to y = \min(1 - x + y, 1)$), the *Goguen* (*product*) *structure* ($x \otimes y = x \cdot y$, $x \to y = 1$ if $x \leqslant y$ and $= y/x$ otherwise) and the *Gödel structure* ($x \otimes y = \min(x, y)$, $x \to y = 1$ if $x \leqslant y$ and $= y$ otherwise). Another important set of truth values is the set $\{a_0, a_1, \dots, a_n\}$, $0 = a_0 < \dots < a_n = 1$, with $a_k \otimes a_l = a_{\max(k+l-n,0)}$ and $a_k \to a_l = a_{\min(n-k+l,n)}$. A special case of the latter algebras is the two-element Boolean algebra of classical logic with the support $\{0, 1\}$. The only adjoint pair on the two-element Boolean algebra consists of the classical conjunction and implication operations. This structure of truth values is called the *Boolean structure*.

In the sequel $\mathcal{L}$ will be a complete residuated lattice. A *fuzzy subset* of a set A *over* $\mathcal{L}$, or simply a *fuzzy subset* of A, is any mapping from A into L. Ordinary crisp subsets of A are considered as fuzzy subsets of A taking membership values in the set $\{0, 1\} \subseteq L$. Let f and g be two fuzzy subsets of A. The *equality* of f and g is defined as the usual equality of mappings, i.e., $f = g$ if and only if $f(x) = g(x)$, for every $x \in A$. The *inclusion* $f \leqslant g$ is also defined pointwise: $f \leqslant g$ if and only if $f(x) \leqslant g(x)$, for every $x \in A$. Endowed with this partial order the set L^A of all fuzzy subsets of A forms a complete residuated lattice, in which the meet (intersection) $\bigwedge_{i \in I} f_i$ and the join (union) $\bigvee_{i \in I} f_i$ of an arbitrary family $\{f_i\}_{i \in I}$ of fuzzy subsets of A are mappings from A into L defined by

$$\left(\bigwedge_{i \in I} f_i\right)(x) = \bigwedge_{i \in I} f_i(x), \qquad \left(\bigvee_{i \in I} f_i\right)(x) = \bigvee_{i \in I} f_i(x),$$

for all $x \in A$. The *height* $\|f\|$ of a fuzzy set $f \in L^A$ is defined by

$$\|f\| = \bigvee_{a \in A} f(a).$$

A *fuzzy relation* between sets A and B (in this order) is any mapping from $A \times B$ to L, i.e. , any fuzzy subset of $A \times B$, and the equality, inclusion (ordering), joins and meets of fuzzy relations are defined as for fuzzy sets. Set of all fuzzy relations between A and B will be denoted by $L^{A \times B}$. In particular, a fuzzy relation on a set A is any function from $A \times A$ to L, i.e., any fuzzy subset of $A \times A$. The set of all fuzzy relations on A will be denoted by $L^{A \times A}$. The *reverse* or *inverse* of a fuzzy relation $\alpha \in L^{A \times B}$ is a fuzzy relation $\alpha^{-1} \in L^{B \times A}$ defined by $\alpha^{-1}(b, a) = \alpha(a, b)$, for all $a \in A$ and $b \in B$. A crisp relation is a fuzzy relation which takes values only in the set $\{0, 1\}$, and if α is a crisp relation of A to B, then expressions "$\alpha(a, b) = 1$" and "$(a, b) \in \alpha$" will have the same meaning.

For non-empty sets A, B and C, and fuzzy relations $\alpha \in L^{A \times B}$ and $\beta \in L^{B \times C}$, their *composition* $\alpha \circ \beta \in L^{A \times C}$ is a fuzzy relation defined by

$$(\alpha \circ \beta)(a, c) = \bigvee_{b \in B} \alpha(a, b) \otimes \beta(b, c), \tag{2.5}$$

for all $a \in A$ and $c \in C$. For $f \in L^A$, $\alpha \in L^{A \times B}$ and $g \in L^B$, compositions $f \circ \alpha \in L^B$ and $\alpha \circ g \in L^A$ are fuzzy sets defined by

$$(f \circ \alpha)(b) = \bigvee_{a \in A} f(a) \otimes \alpha(a, b), \qquad (\alpha \circ g)(a) = \bigvee_{b \in B} \alpha(a, b) \otimes g(b), \tag{2.6}$$

for every $a \in A$ and $b \in B$. Finally, the composition of two fuzzy sets $f, g \in L^A$ is an element $f \circ g \in L$ (scalar) defined by

$$f \circ g = \bigvee_{a \in A} f(a) \otimes g(a). \tag{2.7}$$

When the underlying sets are finite, fuzzy relations can be interpreted as matrices and fuzzy sets as vectors with entries in L, and then the composition of fuzzy relations can be interpreted as the matrix product, compositions of fuzzy sets and fuzzy relations as vector-matrix products, and the composition of two fuzzy set as the scalar (dot) product.

It is easy to verify that the composition of fuzzy relations is associative, i.e.,

$$(\alpha \circ \beta) \circ \gamma = \alpha \circ (\beta \circ \gamma), \tag{2.8}$$

for all $\alpha \in L^{A\times B}$, $\beta \in L^{B\times C}$ and $\gamma \in L^{C\times D}$, and

$$(f \circ \alpha) \circ \beta = f \circ (\alpha \circ \beta), \quad (f \circ \alpha) \circ g = f \circ (\alpha \circ g), \quad (\alpha \circ \beta) \circ h = \alpha \circ (\beta \circ h) \tag{2.9}$$

for all $\alpha \in L^{A\times B}$, $\beta \in L^{B\times C}$, $f \in L^A$, $g \in L^B$ and $h \in L^C$. Hence, all parentheses in (2.8) and (2.9) can be omitted.

2.2. Fuzzy automata and languages

Throughout this paper, $\mathbb{N}$ denotes the set of natural numbers (without zero), X is an (finite) alphabet, X^+ and X^* denote, respectively, the free semigroup and the free monoid over X, ε denotes the empty word in X^*, and if not noted otherwise, $\mathcal{L}$ is a complete residuated lattice.

A *fuzzy automaton over* $\mathcal{L}$ and X, or simply a *fuzzy automaton*, is a quadruple $\mathcal{A} = (A, \sigma, \delta, \tau)$, where A is a non-empty set, called the *set of states*, $\delta : A \times X \times A \to L$ is a fuzzy subset of $A \times X \times A$, called the *fuzzy transition function*, and $\sigma : A \to L$ and $\tau : A \to L$ are fuzzy subsets of A, called the *fuzzy set of initial states* and the *fuzzy set terminal states*, respectively. We can interpret $\delta(a, x, b)$ as the degree to which an input letter $x \in X$ causes a transition from a state $a \in A$ into a state $b \in A$, and we can interpret $\sigma(a)$ and $\tau(a)$ as the degrees to which a is respectively an input state and a terminal state. For methodological reasons we allow the set of states A to be infinite. A fuzzy automaton whose set of states is finite is called a *fuzzy finite automaton*. A fuzzy automaton over the Boolean structure is called a *nondeterministic automaton* or a *Boolean automaton*.

We can visualize a fuzzy finite automaton $\mathcal{A} = (A, \sigma, \delta, \tau)$ representing it as a labelled directed graph whose nodes are states of $\mathcal{A}$, an edge from a node a to a node b is labelled by pairs of the form $x/\delta_x(a, b)$, for any $x \in X$, and for any node a we draw an arrow labelled by $\sigma(a)$ that enters this node, and an arrow labelled by $\tau(a)$ coming out of this node. For the sake of simplicity, we do not draw edges whose all labels are of the form $x/0$, and incoming and outgoing arrows labelled by 0. In particular, if $\mathcal{A}$ is a Boolean automaton, instead of any label of the form $x/1$ we write just x, initial states are marked by incoming arrows without any label, and terminal states are marked by double circles.

Define a family $\{\delta_x\}_{x\in X}$ of fuzzy relations on A by $\delta_x(a,b) = \delta(a,x,b)$, for each $x \in X$, and all $a,b \in A$, and extend this family to the family $\{\delta_u\}_{u\in X^*}$ inductively, as follows: $\delta_\varepsilon = \Delta_A$, where Δ_A is the crisp equality relation on A, and

$$\delta_{x_1x_2\dots x_n} = \delta_{x_1} \circ \delta_{x_2} \circ \cdots \circ \delta_{x_n} \tag{2.10}$$

for all $n \in \mathbb{N}$, $x_1, x_2, \dots, x_n \in X$. Members of this family are called *fuzzy transiton relations* of $\mathcal{A}$. Evidently, $\delta_{uv} = \delta_u \circ \delta_v$, for all $u, v \in X^*$. In addition, define families $\{\sigma_u\}_{u\in X^*}$ and $\{\tau_u\}_{u\in X^*}$ of fuzzy subsets of A by

$$\sigma_u = \sigma \circ \delta_u, \qquad \tau_u = \delta_u \circ \tau, \tag{2.11}$$

for all $u \in X^*$.

A *fuzzy language* in X^* over $\mathcal{L}$, or just a *fuzzy language*, is any fuzzy subset of X^*, i.e., any function from X^* into L. A *fuzzy language recognized by a fuzzy automaton* $\mathcal{A} = (A, \sigma, \delta, \tau)$ is a fuzzy language $[\![\mathcal{A}]\!] \in L^{X^*}$ defined by

$$[\![\mathcal{A}]\!](u) = \bigvee_{a,b\in A} \sigma(a) \otimes \delta_u(a,b) \otimes \tau(b) = \sigma \circ \delta_u \circ \tau, \tag{2.12}$$

for any $u \in X^*$. In other words, the membership degree of the word u to the fuzzy language $[\![\mathcal{A}]\!]$ is equal to the degree to which $\mathcal{A}$ recognizes or accepts the word u. Fuzzy automata $\mathcal{A}$ and $\mathcal{B}$ are called *language equivalent*, or just *equivalent*, if $[\![\mathcal{A}]\!] = [\![\mathcal{B}]\!]$.

Another kind of fuzzy languages associated with fuzzy automata, which plays an important role in study of fuzzy discrete event systems, is the following one. The *fuzzy language* $[\![\mathcal{A}]\!]_g$ *generated by a fuzzy automaton* $\mathcal{A} = (A, \delta, \sigma, \tau)$ is defined by

$$[\![\mathcal{A}]\!]_g(u) = \|\sigma_u\| = \bigvee_{a,b\in A} \sigma(a) \otimes \delta_u(a,b) = \bigvee_{b\in A} \sigma_u(b), \tag{2.13}$$

for every $u \in X^*$. Intuitively, $[\![\mathcal{A}]\!]_g(u)$ represents the degree to which the input word u causes a transition from some initial state to any other state. Two fuzzy automata $\mathcal{A}$ and $\mathcal{B}$ are called *completely language-equivalent* if $[\![\mathcal{A}]\!] = [\![\mathcal{B}]\!]$ and $[\![\mathcal{A}]\!]_g = [\![\mathcal{B}]\!]_g$.

The *prefix-closure* of a fuzzy language $f \in L^{X^*}$ is a fuzzy language $\bar{f} \in L^{X^*}$ defined by

$$\bar{f}(u) = \bigvee_{v\in X^*} f(uv), \tag{2.14}$$

for each $u \in X^*$. It is easy to verify that the mapping $f \mapsto \bar{f}$ is a closure operator on L^{X^*}, i.e., for arbitrary $f, f_1, f_2 \in L^{X^*}$ we have that

$$f \leqslant \bar{f}, \quad \bar{\bar{f}} = \bar{f} \ \text{ and } \ f_1 \leqslant f_2 \ \text{ implies } \ \bar{f_1} \leqslant \bar{f_2}. \tag{2.15}$$

A fuzzy language $f \in L^{X^*}$ is called *prefix-closed* if $f = \bar{f}$. Clearly, f is prefix-closed if and only if

$$f(uv) \leqslant f(u), \tag{2.16}$$

for all $u, v \in X^*$. In other words, f is prefix-closed if and only if it is a decreasing function from X^* into L (with respect to the prefix order on X^*).

Theorem 2.1. *If a fuzzy language is generated by a fuzzy automaton, then it is prefix-closed.*

Proof. Let $\mathcal{A} = (A, \delta, \sigma, \tau)$ be a fuzzy automaton and $f = [\![\mathcal{A}]\!]_g$. In view of (2.15), we have to prove that $\bar{f} \leqslant f$.

For arbitrary $u, v \in X^*$ we have that

$$\begin{aligned} f(uv) = [\![\mathcal{A}]\!]_g(uv) &= \bigvee_{b \in A} \sigma_{uv}(b) = \bigvee_{b \in A} (\sigma_u \circ \delta_v)(b) = \bigvee_{b \in A} \left(\bigvee_{a \in A} \sigma_u(a) \otimes \delta_v(a, b) \right) \\ &= \bigvee_{a \in A} \left(\sigma_u(a) \otimes \left(\bigvee_{b \in A} \delta_v(a, b) \right) \right) \leqslant \bigvee_{a \in A} \sigma_u(a) = [\![\mathcal{A}]\!]_g(u) = f(u), \end{aligned}$$

and we conclude that

$$\bar{f}(u) = \bigvee_{v \in X^*} f(uv) \leqslant f(u),$$

for each $u \in X^*$. Therefore, f is a prefix-closed fuzzy language. □

3. Automata with fuzzy states

Let $\mathcal{A} = (A, \sigma, \delta, \tau)$ be a fuzzy automaton over X and $\mathcal{L}$. The fuzzy transition function δ is called *crisp-deterministic* if for every $x \in X$ and every $a \in A$ there exists $a' \in A$ such that $\delta_x(a, a') = 1$, and $\delta_x(a, b) = 0$, for all $b \in A \setminus \{a'\}$. The fuzzy set of initial states σ is called *crisp-deterministic* if there exists $a_0 \in A$ such that $\sigma(a_0) = 1$, and $\sigma(a) = 0$, for every $a \in A \setminus \{a_0\}$. If both σ and δ are crisp-deterministic, then $\mathcal{A}$ is called a *crisp-deterministic fuzzy automaton* (for short: *cdfa*), and if it is finite, then it is called a *crisp-deterministic fuzzy finite automaton* (for short: *cdffa*).

A crisp-deterministic fuzzy automaton can be equivalently defined as a quadruple $\mathcal{A} = (A, a_0, \delta, \tau)$, where A is a non-empty *set of states*, $\delta : A \times X \to A$ is a *transition function*, $a_0 \in A$ is an *initial state* and $\tau \in L^A$ is a *fuzzy set of terminal states*. The transition function δ can be extended to a function $\delta^* : A \times X^* \to A$ in the following way: $\delta^*(a, \varepsilon) = a$, for every $a \in A$, and $\delta^*(a, ux) = \delta(\delta^*(a, u), x)$, for all $a \in A$, $u \in X^*$ and $x \in X$. To simplify the notation, and without danger of confusion, we will write δ instead of δ^*. Note that the initial state and transitions of a crisp-deterministic fuzzy automaton are graphically represented as in the case of Boolean automata, and the fuzzy set of terminal states is represented as in the case of fuzzy finite automata.

The fuzzy language $[\![\mathcal{A}]\!]$ recognized by a crisp-deterministic fuzzy automaton $\mathcal{A} = (A, a_0, \delta, \tau)$ is given by

$$[\![\mathcal{A}]\!](u) = \tau(\delta(a_0, u)), \tag{3.1}$$

for every $u \in X^*$. Clearly, the image of $[\![\mathcal{A}]\!]$ is contained in the image of τ which is finite if the set of states A is finite. A fuzzy language $f : X^* \to L$ is called *cdffa-recognizable* if there is a crisp-deterministic fuzzy finite automaton $\mathcal{A}$ over X and $\mathcal{L}$ such that $[\![\mathcal{A}]\!] = f$. Then we say that $\mathcal{A}$ *recognizes* f.

A state $a \in A$ is called *accessible* if there exists $u \in X^*$ such that $\delta^*(a_0, u) = a$. If every state of $\mathcal{A}$ is accessible, then $\mathcal{A}$ is called an *accessible crisp-deterministic fuzzy automaton*.

The concept of recognition of a fuzzy language by a crisp-deterministic fuzzy automaton is given in a very elegant way, but the concept of the fuzzy language generated by such an automaton does not make much sense. Namely, if we consider a crisp-deterministic fuzzy automaton $\mathcal{A}$ as a fuzzy automaton, then the fuzzy language generated by $\mathcal{A}$ is X^*. Even in constructions where a fuzzy automaton $\mathcal{A}$ is converted into a language-equivalent crisp-deterministic fuzzy automaton which states are some particular fuzzy subsets of the set of states of $\mathcal{A}$, the fuzzy nature of these states is taken into account only in the construction of the fuzzy set of terminal states and nowhere else.

A model of automata with fuzzy states which takes into account the fuzzy nature of these states has been used in [29–31, 42, 56], in the study of fuzzy discrete event systems. Here we provide a more general definition. An *automaton with fuzzy states* (or *fuzzy automaton with fuzzy states*, or *fuzzy-state automaton*) is defined as a quintuple $\bar{\mathcal{A}} = (\bar{A}, \sigma, \bar{\delta}, \bar{\tau})$, where A is a non-empty set, called the *set of states*, $\bar{A} \subseteq L^A$ is the *set of fuzzy states* (not necessarily finite), $\bar{\delta} : \bar{A} \times X \to \bar{A}$ is the *transition function*, and $\sigma \in \bar{A}$ is the *fuzzy initial state*, and $\bar{\tau} \in L^{\bar{A}}$ is the *fuzzy terminal state*. If the set $\bar{A}$ is finite, then $\bar{\mathcal{A}}$ is called a *finite automaton with fuzzy states*. The function $\bar{\delta}$ can be extended to a function $\bar{\delta}^* : \bar{A} \times X^* \to \bar{A}$, as follows: for all $\alpha \in \bar{A}$, $u \in X^*$ and $x \in X$ we set $\bar{\delta}^*(\alpha, \varepsilon) = \alpha$ and $\bar{\delta}^*(\alpha, ux) = \bar{\delta}(\bar{\delta}^*(\alpha, u), x)$. To simplify the notation, and without danger of confusion, we will write $\bar{\delta}$ instead of $\bar{\delta}^*$.

The *fuzzy language recognized by an automaton with fuzzy states* $\bar{\mathcal{A}} = (\bar{A}, \sigma, \bar{\delta}, \bar{\tau})$, denoted by $[\![\bar{\mathcal{A}}]\!]$, is a fuzzy language in L^{X^*} defined by

$$[\![\bar{\mathcal{A}}]\!](u) = \bar{\tau}(\bar{\delta}(\sigma, u)), \tag{3.2}$$

for each $u \in X^*$. On the other hand, the *fuzzy language generated by* $\bar{\mathcal{A}}$, denoted by $[\![\bar{\mathcal{A}}]\!]_g$, is defined by

$$[\![\bar{\mathcal{A}}]\!]_g(u) = \|\bar{\delta}(\sigma, u)\| = \bigvee_{a \in A} \sigma'(a), \tag{3.3}$$

for each $u \in X^*$ and $\sigma' = \bar{\delta}(\sigma, u)$.

Note that automata studied in [29–31, 42, 56] are a special case of the above defined automata. The set of fuzzy states of these automata is $\bar{A} = [0,1]^A$, the transition function is defied by means of a given family $\{\delta_x\}_{x\in X}$ of fuzzy relations on A indexed by the alphabet X, and the fuzzy terminal state $\bar{\tau}$ is defined by means of a given set of fuzzy states $T \subseteq \bar{A}$, in the following way:

$$\bar{\delta}(\alpha, x) = \alpha \circ \delta_x, \qquad \bar{\tau}(\alpha) = \bigvee_{\beta\in T} \alpha \circ \beta, \tag{3.4}$$

for all $\alpha \in \bar{A}$ and $x \in X$. An automaton with fuzzy states defined in this way will be called *compositionally defined*. Let us observe that the same definition makes sense if we take $\bar{A}$ to be any subset of $[0,1]^A$ closed under compositions with δ_x, for all $x \in X$ (in this case $\bar{A}$ may be finite). Moreover,

$$\bar{\tau}(\alpha) = \bigvee_{\beta\in T} \alpha \circ \beta = \alpha \circ \bigvee_{\beta\in T} \beta,$$

and without loss of generality we can assume that T consists of a single fuzzy subset of A. It is also worth noting that the first equality in (3.4) is equivalent to the the following one

$$\bar{\delta}(\alpha, u) = \alpha \circ \delta_u \tag{3.5}$$

for every $\alpha \in \bar{A}$ and $u \in X^+$, where for an arbitrary $u = x_1x_2\dots x_n \in X^+$ a fuzzy relation δ_u stands for a composition $\delta_{x_1} \circ \delta_{x_2} \circ \dots \delta_{x_n}$.

If $\mathcal{A}$ is a fuzzy automaton and $\bar{\mathcal{A}}$ is an automaton with fuzzy states, then we say that $\mathcal{A}$ and $\bar{\mathcal{A}}$ are *language-equivalent* if $[\![\mathcal{A}]\!] = [\![\bar{\mathcal{A}}]\!]$, and *completely language-equivalent* if $[\![\mathcal{A}]\!] = [\![\bar{\mathcal{A}}]\!]$ and $[\![\mathcal{A}]\!]_g = [\![\bar{\mathcal{A}}]\!]_g$.

The first theorem of this section shows how a crisp-deterministic automaton can be transformed into a language-equivalent automaton with fuzzy states, and vice versa.

Theorem 3.1. *Let $\mathcal{D} = (D, s, \Delta, \theta)$ be a crisp-deterministic fuzzy automaton, let A be a non-empty set, $\bar{A}$ a family of fuzzy subsets of A, $\phi : \bar{A} \to D$ a bijective function, and let $\bar{\delta} : \bar{A} \times X \to \bar{A}$, $\sigma \in \bar{A}$, and $\bar{\tau} \in L^{\bar{A}}$ be defined by*

$$\bar{\delta}(\alpha, x) = \phi^{-1}(\Delta(\phi(\alpha), x)), \quad \sigma = \phi^{-1}(s), \quad \bar{\tau}(\alpha) = \theta(\phi(\alpha)),$$

for all $\alpha \in \bar{A}$ and $x \in X$. Then $\bar{\mathcal{A}} = (\bar{A}, \sigma, \bar{\delta}, \bar{\tau})$ is an automaton with fuzzy states which is language-equivalent to $\mathcal{D}$.

Conversely, if $\bar{\mathcal{A}} = (\bar{A}, \sigma, \bar{\delta}, \bar{\tau})$ is an automaton with fuzzy states, and if we consider $\bar{A}$ as an ordinary set, then $\bar{\mathcal{A}}$ becomes a crisp-deterministic fuzzy automaton which is language-equivalent to $\bar{\mathcal{A}}$.

Proof. It is clear that the functions $\bar{\delta}$ and $\bar{\tau}$ are well-defined, and $\bar{\mathcal{A}} = (\bar{A}, \sigma, \bar{\delta}, \bar{\tau})$ is an automaton with fuzzy states. Also, it can be easily verified that $\bar{\delta}(\alpha, u) = \phi^{-1}(\Delta(\phi(\alpha), u))$, for all $\alpha \in \bar{A}$ and $u \in X^*$.

Furthermore, for every $u \in X^*$ we have that

$$[\![\bar{\mathcal{A}}]\!](u) = \bar{\tau}(\bar{\delta}(\sigma, u)) = \theta(\phi(\phi^{-1}(\Delta(\phi(\phi^{-1}(s)), u)))) = \theta(\Delta(s, u)) = [\![\mathcal{D}]\!](u),$$

and therefore, $\bar{\mathcal{A}}$ and $\mathcal{D}$ are language-equivalent.

The converse is clear, because the fuzzy nature of the states from $\bar{A}$ does not affect the recognition of a language. □

Nerode automata were introduced in [14, 15] as a means for determinization of fuzzy automata, and in [4,17] they were used in determinization of weighted finite automata over strong bimonoids. Originally, Nerode automata were defined to be crisp-deterministic fuzzy (or weighted) automata, but from their construction we can see that in essence they are automata with fuzzy states. In the sequel, we define Nerode automata as automata with fuzzy states.

Let $\mathcal{A} = (A, \sigma, \delta, \tau)$ be a fuzzy automaton. Let us set $\bar{A}_N = \{\sigma_u \mid u \in X^*\}$, and define $\bar{\delta}_N : \bar{A}_N \times X \to \bar{A}_N$ and $\bar{\tau}_N : \bar{A}_N \to L$ by

$$\bar{\delta}_N(\sigma_u, x) = \sigma_{ux} \quad \text{and} \quad \bar{\tau}_N(\sigma_u) = \sigma_u \circ \tau, \tag{3.6}$$

for every $u \in X^*$ and $x \in X$. Then $\bar{\mathcal{A}}_N = (\bar{A}_N, \sigma_\varepsilon, \bar{\delta}_N, \bar{\tau}_N)$ is an automaton with fuzzy states and we will call it the *Nerode automaton* of the fuzzy automaton $\mathcal{A}$. Let us note that $\sigma_\varepsilon = \sigma$.

As shown in [14], the Nerode automaton of a fuzzy automaton $\mathcal{A}$ is language-equivalent to $\mathcal{A}$. Here we prove that they are completely language-equivalent.

Theorem 3.2. *For each fuzzy automaton $\mathcal{A}$, the Nerode automaton $\bar{\mathcal{A}}_N$ of $\mathcal{A}$ is completely language-equivalent to $\mathcal{A}$.*

Proof. For an arbitrary $u \in X^*$ we have that

$$[\![\bar{\mathcal{A}}_N]\!](u) = \bar{\tau}(\bar{\delta}_N(\sigma, u)) = \sigma_u \circ \tau = \sigma \circ \delta_u \circ \tau = [\![\mathcal{A}]\!](u),$$

and

$$[\![\bar{\mathcal{A}}_N]\!]_g(u) = \|\bar{\delta}_N(\sigma, u)\| = \|\sigma_u\| = [\![\mathcal{A}]\!]_g(u).$$

Therefore, $\bar{\mathcal{A}}_N$ and $\mathcal{A}$ are completely language equivalent. □

An automaton with fuzzy states $\bar{\mathcal{A}} = (\bar{A}, \sigma, \bar{\delta}, \bar{\tau})$ is called *accessible* if for every $\alpha \in \bar{A}$ there exists $u \in X^*$ such that $\alpha = \bar{\delta}(\sigma, u)$, i.e., if every state can be reached from the initial state.

The next theorem provides necessary and sufficient conditions under which an automaton with fuzzy states is compositionally defined or it can be represented as the Nerode automaton of some fuzzy automaton.

Theorem 3.3. *An automaton $\bar{\mathcal{A}} = (\bar{A}, \sigma, \bar{\delta}, \bar{\tau})$ with fuzzy states is compositionally defined if and only if the following is true:*

(i) *For every $x \in X$ there exists at least one solution to the system of equations*

$$\alpha \circ R_x = \bar{\delta}(\alpha, x), \qquad \alpha \in \bar{A}, \tag{3.7}$$

where R_x is an unknown taking values in $L^{A\times A}$.

(ii) *There exists at least one solution to the system of equations*

$$\alpha \circ U = \bar{\tau}(\alpha), \qquad \alpha \in \bar{A}, \tag{3.8}$$

where U is an unknown taking values in L^{A}.

In addition, $\bar{\mathcal{A}}$ can be represented as the Nerode automaton of some fuzzy automaton $\mathcal{A} = (A, \sigma, \delta, \tau)$ if and only if it is accessible and (i) *and* (ii) *hold.*

Proof. For any $x \in X$ let $\delta_x \in L^{A\times A}$ be an arbitrary solution to (3.7), i.e., let $\alpha \circ \delta_x = \bar{\delta}(\alpha, x)$, for each $\alpha \in \bar{A}$, and define a function $\delta : A\times X\times A \to L$ by $\delta(a, x, b) = \delta_x(a, b)$, for all $a, b \in A$ and $x \in X$. Moreover, let $\tau \in L^A$ be an arbitrary solution to (3.8), i.e., let $\alpha \circ \tau = \bar{\tau}(\alpha)$, for each $\alpha \in \bar{A}$. Then $\bar{\mathcal{A}}$ is a compositionally defined automaton with fuzzy states determined by the family $\{\delta_x\}_{x\in X}$ and the set $T = \{\tau\}$. The converse is evident.

Next, let $\bar{\mathcal{A}}$ be compositionally defined and accessible and let $\mathcal{A} = (A, \sigma, \delta, \tau)$ be a fuzzy automaton, where fuzzy sets δ and τ are defined in the observation above. Then, by the accessibility of $\bar{\mathcal{A}}$ and (3.5) one conclude that there exists $\sigma \in \bar{A}$ such that for each $\alpha \in \bar{A}$ the following is true:

$$\alpha = \bar{\delta}(\sigma, u) = \sigma \circ \delta_u = \sigma_u,$$

which yields $\bar{A} = \{\sigma_u \mid u \in X^*\} = \bar{A}_N$. Also, for all $u \in X^*$ and $x \in X$ we have that

$$\bar{\delta}(\sigma_u, x) = \sigma_u \circ \delta_x = \sigma_{ux} = \bar{\delta}_N(\sigma_u, x), \quad \bar{\tau}(\sigma_u) = \sigma_u \circ \tau = \bar{\tau}_N(\sigma_u).$$

Therefore, $\bar{\mathcal{A}} = \bar{\mathcal{A}}_N$. The converse is evident.

The assertion concerning the Nerode automata is an immediate consequence of the fact that an automaton with fuzzy states can be represented as the Nerode automaton of some fuzzy automaton if and only if it is compositionally defined and accessible. □

Now we give an example which shows that there are automata with fuzzy states which can not be represented as the Nerode automaton of some fuzzy automaton. i.e., which is not compositionally defined.

Example 3.1. Let $\mathcal{L}$ be the Gödel structure, $X = \{x, y\}$, and let $\bar{\mathcal{A}} = (\bar{A}, \alpha_0, \bar{\delta}, \bar{\tau})$ be an automaton with fuzzy states given by the transition graph shown in Figure 3.1, where $|A| = 2$, $\bar{A} = \{\alpha_0, \alpha_1, \alpha_2\}$, and

$$\alpha_0 = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad \alpha_1 = \begin{bmatrix} 0 & 1 \end{bmatrix}, \quad \alpha_2 = \begin{bmatrix} 1 & 1 \end{bmatrix}, \quad \bar{\tau} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

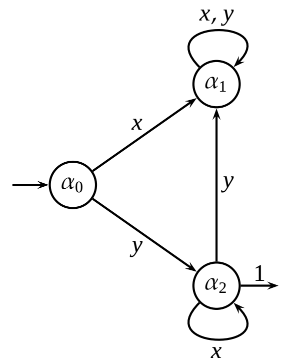

FIG. 3.1: The graph of the automaton with fuzzy states $\bar{\mathcal{A}}$ from Example 3.1.

It is easy to check that systems

$$\begin{array}{lll} \alpha_0 \circ R_x = \alpha_1 & \alpha_0 \circ R_y = \alpha_2 & \alpha_0 \circ U = \bar{\tau}(\alpha_0) \\ \alpha_1 \circ R_x = \alpha_1 & \alpha_1 \circ R_y = \alpha_1 & \alpha_1 \circ U = \bar{\tau}(\alpha_1) \\ \alpha_2 \circ R_x = \alpha_2 & \alpha_2 \circ R_y = \alpha_1 & \alpha_2 \circ U = \bar{\tau}(\alpha_2) \end{array}$$

where R_x and R_y are unknown fuzzy relations on A and U is an unknown fuzzy subset of A, do not have solutions. Therefore, $\bar{\mathcal{A}}$ is not compositionally defined and it can not be represented as the Nerode automaton of some fuzzy automaton with A as its set of states.

In addition to the Nerode automaton, there are also other types of automata with fuzzy states that can be constructed starting from a given fuzzy automaton. We will mention two automata of that kind. The first automaton has been defined in [18] by means of a fuzzy relation φ on the set of states of a fuzzy automaton $\mathcal{A} = (A, \sigma, \delta, \tau)$ in the following way: the set of fuzzy states $\bar{A}_\varphi = \{\varphi_u \mid u \in X^*\}$ is defined inductively by

$$\varphi_\varepsilon = \sigma \circ \varphi, \qquad \varphi_{ux} = \varphi_u \circ \delta_x \circ \varphi,$$

for all $u \in X^*$ and $x \in X$, and $\bar{\delta}_\varphi : \bar{A}_\varphi \times X \to \bar{A}_\varphi$ and $\bar{\tau}_\varphi : \bar{A}_\varphi \to L$ are defined by

$$\bar{\delta}_\varphi(\varphi_u, x) = \varphi_{ux}, \qquad \bar{\tau}_\varphi(\varphi_u) = \varphi_u \circ \tau,$$

for all $\varphi_u \in \bar{A}_\varphi$ and $x \in X$. If φ is reflexive and a *weakly right invariant* fuzzy relation on $\mathcal{A}$, i.e., if it satisfies $\varphi \circ \tau_u \leqslant \tau_u$, for every $u \in X^*$, then the automaton with fuzzy states $\bar{\mathcal{A}}_\varphi = (\bar{A}_\varphi, \varphi_\varepsilon, \bar{\delta}_\varphi, \bar{\tau}_\varphi)$ is language-equivalent to $\mathcal{A}$.

Another way to convert a given fuzzy automaton $\mathcal{A} = (A, \sigma, \delta, \tau)$ into a language-equivalent automaton with fuzzy states has been proposed in [32]. The set of fuzzy states of this automaton is $\bar{A}_d = \{d_u \mid u \in X^*\}$, which is also defined inductively by

$$d_\varepsilon(a) = \bigwedge_{w \in X^*} \tau_w(a) \to \sigma \circ \tau_w,$$

and

$$d_{ux}(a) = \bigwedge_{w\in X^*} \tau_w(a) \to d_u \circ \delta_x \circ \tau_w = \bigwedge_{w\in X^*} \tau_w(a) \to d_u \circ \tau_{xw},$$

for all $a \in A$, $u \in X^*$ and $x \in X$, and $\bar{\delta}_d : \bar{A}_d \times X \to \bar{A}_d$ and $\bar{\tau}_d : \bar{A}_d \to L$ are defined by

$$\bar{\delta}_d(d_u, x) = d_{ux}, \qquad \bar{\tau}_d(d_u) = d_u \circ \tau,$$

for all $d_u \in \bar{A}_d$ and $x \in X$. The automaton with fuzzy states $\bar{\mathcal{A}}_d = (\bar{A}_d, d_\varepsilon, \bar{\delta}_d, \bar{\tau}_d)$ is language-equivalent to $\mathcal{A}$, and according to Theorem 3.1 and Theorem 3.3 [32], it is a minimal automaton with fuzzy states language-equivalent to $\mathcal{A}$.

An interesting open problem is the following one:

Problem 3.1. Whether the above mentioned automata $\bar{A}_\varphi$ and $\bar{A}_d$ are completely language-equivalent with the original fuzzy automaton $\mathcal{A}$ (as is the Nerode automaton $\bar{\mathcal{A}}_N$)?

Another kind of fuzzy automata which can be considered as automata with fuzzy states are the derivative automata introduced in [15].

For a fuzzy language $f \in L^{X^*}$ and $u \in X^*$, a fuzzy language $f_u \in L^{X^*}$ defined by

$$f_u(v) = f(uv), \tag{3.9}$$

for each $v \in X^*$, is called the *derivative* of f with respect to u. In particular, if f is a crisp language, then its derivative with respect to u is the crisp language $f_u = \{v \in X^* \mid uv \in f\}$. Derivatives of crisp languages are also known as *right quotients*, *quotients* or *residuals* of languages.

Let $\bar{A}_f = \{f_u \mid u \in X^*\}$ be the set of all derivatives of f, and let $\bar{\delta}_f : \bar{A}_f \times X \to \bar{A}_f$ and $\bar{\tau}_f \in L^{\bar{A}_f}$ be defined by

$$\bar{\delta}_f(g, x) = g_x, \qquad \bar{\tau}_f(g) = g(\varepsilon), \tag{3.10}$$

for all $g \in \bar{A}_f$ and $x \in X$. We have the following:

Theorem 3.4. *For any fuzzy language $f \in L^{X^*}$ we have that $\bar{\mathcal{A}}_f = (\bar{A}_f, f, \bar{\delta}_f, \bar{\tau}_f)$ is an automaton with fuzzy states.*

Moreover, $\bar{\mathcal{A}}_f$ is a minimal automaton with fuzzy states which recognizes f and generates the prefix-closure $\bar{f}$ of f.

Proof. According to Theorem 3.1 and Theorem 4.1 [15], $\bar{\mathcal{A}}_f$ is a minimal automaton with fuzzy states which recognizes f. Furthermore, for an arbitrary $u \in X^*$ we have that

$$[\![\bar{\mathcal{A}}_f]\!]_g(u) = \|\bar{\delta}^*_f(f, u)\| = \|f_u\| = \bigvee_{v\in X^*} f_u(v) = \bigvee_{v\in X^*} f(uv) = \bar{f}(u),$$

and hence, $[\![\bar{\mathcal{A}}_f]\!]_g = \bar{f}$.

Consequently, $\bar{\mathcal{A}}_f$ is a minimal automaton with fuzzy states which both recognizes f and generates $\bar{f}$. □

The automaton $\bar{\mathcal{A}}_f = (\bar{A}_f, f, \bar{\delta}_f, \bar{\tau}_f)$ is called the *derivative automaton* of the fuzzy language f.

Corollary 3.1. *If a fuzzy language is prefix-closed, then it is generated by an automaton with fuzzy states.*

Proof. If a fuzzy language $f \in L^{X^*}$ is prefix-closed, then it is generated by $\bar{\mathcal{A}}_f$, in view of Theorem 3.4. □

The converse of the previous corollary does not necessarily hold. The next example shows that there is a fuzzy language which is generated by an automaton with fuzzy states, but it is not prefix-closed.

Example 3.2. Let $\mathcal{L}$ be the Gödel structure and $X = \{x, y\}$, and let $\bar{\mathcal{A}} = (\bar{A}, \alpha_0, \bar{\delta}, \bar{\tau})$ be an automaton with fuzzy states given by the transition graph shown in Figure 3.1, where $|A| = 2$, $\bar{A} = \{\alpha_0, \alpha_1, \alpha_2\}$, and $\bar{\tau}$ is an arbitrary fuzzy subset of $\bar{A}$. For the sake of simplicity set $f = [\![\bar{\mathcal{A}}]\!]_g$.

We have that $f(y) = \|\alpha_2\|$, $f(y^2) = \|\alpha_1\|$, and we can easily choose $\alpha_1, \alpha_2 \in L^A$ such that $\|\alpha_1\| > \|\alpha_2\|$. Then $f(y^2) > f(y)$, and hence, f is not a prefix-closed fuzzy language.

Finally, we prove the following theorem.

Theorem 3.5. *Let $f \in L^{X^*}$ be a fuzzy language. Then:*

(i) *For every $x \in X$ there exists at least one solution to the system of equations*

$$g \circ R_x = g_x, \qquad g \in \bar{A}_f, \tag{3.11}$$

where R_x is an unknown taking values in $L^{X^* \times X^*}$.

(ii) *There exists at least one solution to the system of equations*

$$g \circ U = g(\varepsilon), \qquad g \in \bar{A}_f, \tag{3.12}$$

where U is an unknown taking values in L^{X^*}.

Consequently, the derivative automaton $\bar{\mathcal{A}}_f$ can be compositionally defined.

Proof. Define a fuzzy relation Q on X^* by

$$Q(u, v) = \bigwedge_{g \in \bar{A}_f} g(u) \to g(v),$$

for all $u, v \in X^*$. It is well-known that Q is the greatest solution to system $g \circ R = g$, $g \in \bar{A}_f$, where R is an unknown fuzzy relation on X^*.

(i) For an arbitrary $x \in X$, define a fuzzy relation Q_x on X^* by $Q_x(u,v) = Q(u,xv)$, for all $u,v \in X^*$. Then for all $g \in \bar{A}_f$ and $v \in X^*$ we have that

$$(g \circ Q_x)(v) = \bigvee_{u \in X^*} g(u) \otimes Q_x(u,v) = \bigvee_{u \in X^*} g(u) \otimes Q(u,xv) = (g \circ Q)(xv) = g(xv) = g_x(v),$$

which yields $g \circ Q_x = g_x$. Therefore, Q_x is a solution to (3.11).

(ii) Define a fuzzy subset V of X^* by $V(u) = Q(u,\varepsilon)$, for each $u \in X^*$. Then for every $g \in \bar{A}_f$ we have that

$$g \circ V = \bigvee_{u \in X^*} g(u) \otimes V(u) = \bigvee_{u \in X^*} g(u) \otimes Q(u,\varepsilon) = (g \circ Q)(\varepsilon) = g(\varepsilon),$$

and hence, V is a solution to (3.12).

It is clear that solvability of the above systems means that $\bar{\mathcal{A}}_f$ can be compositionally defined. □

The following result is an immediate consequence of the previous theorem:

Corollary 3.2. *Let $f \in L^{X^*}$ be a fuzzy language. Then the derivative automaton $\bar{\mathcal{A}}_f$ is the Nerode automaton of some fuzzy automaton $\mathcal{A}$.*

Proof. Since the derivative automaton $\bar{\mathcal{A}}_f$ is compositionally defined, by Theorem 3.3 it is sufficient to prove that $\bar{\mathcal{A}}_f$ is accessible.

For an arbitrary $g \in \bar{A}_f$ there exists $u \in X^*$ such that $g = f_u$, and therefore $\bar{\delta}_f(f,u) = f_u = g$. In conclusion, the derivative automaton $\bar{\mathcal{A}}_f$ is accessible. □

REFERENCES

1. R. Bělohlávek, *Fuzzy Relational Systems: Foundations and Principles*. Kluwer, New York, 2002.
2. R. Bělohlávek, *Determinism and fuzzy automata.* Information Sciences **143** (2002), 205–209.
3. R. Bělohlávek and V. Vychodil, *Fuzzy Equational Logic*. Studies in Fuzziness and Soft Computing, Springer, Berlin-Heidelberg, 2005.
4. M. Ćirić, M. Droste, J. Ignjatović and H. Vogler, *Determinization of weighted finite automata over strong bimonoids*. Information Sciences **180** (2010), 3497–3520.
5. M. Ćirić, J. Ignjatović, N. Damljanović and M. Bašić, *Bisimulations for fuzzy automata*, Fuzzy Sets and Systems **186** (2012), 100–139.
6. M. Ćirić, J. Ignjatović, I. Jančić and N. Damljanović, *Computation of the greatest simulations and bisimulations between fuzzy automata*, Fuzzy Sets and Systems **208** (2012), 22–42.
7. M. Ćirić, A. Stamenković, J. Ignjatović and T. Petković, *Factorization of fuzzy automata*, In: Csuhaj-Varju, E., Ésik, Z. (eds.), FCT 2007, Springer, Heidelberg, Lecture Notes in Computer Science **4639** (2007), 213–225.
8. M. Ćirić, A. Stamenković, J. Ignjatović and T. Petković, *Fuzzy relation equations and reduction of fuzzy automata*, Journal of Computer and System Sciences **76** (2010), 609–633.

9. M. Droste, T. Stüber and H. Vogler, *Weighted finite automata over strong bimonoids*, Information Sciences **180** (2010), 156–166.
10. M. Droste and H. Vogler, *Kleene and Büchi results for weighted automata and multi-valued logics over arbitrary bounded lattices*, in: Y. Gao et al. (eds.), DLT 2010, Lecture Notes in Computer Science, **6224**, 2010, pp. 160–172.
11. D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*, Academic Press, New York, 1980.
12. M. M. Gupta, G. N. Saridis and B. R. Gaines, *Fuzzy Automata and Decision Processes*, North-Holland, New York, 1977.
13. J. Ignjatović and M. Ćirić, *Formal power series and regular operations on fuzzy languages*. Information Sciences **180** (2010), 1104–1120.
14. J. Ignjatović, M. Ćirić and S. Bogdanović, *Determinization of fuzzy automata with membership values in complete residuated lattices*. Information Sciences **178** (2008), 164–180.
15. J. Ignjatović, M. Ćirić, S. Bogdanović and T. Petković, *Myhill-Nerode type theory for fuzzy languages and automata*. Fuzzy Sets and Systems **161** (2010), 1288-1324.
16. Z. Jančić and M. Ćirić, *Brzozowski type determinization for fuzzy automata*. Fuzzy Sets and Systems **249** (2014), 73–82.
17. Z. Jančić, J. Ignjatović and M. Ćirić, *An improved algorithm for determinization of weighted and fuzzy automata*. Information Sciences **181** (2011), 1358–1368.
18. Z. Jančić, I. Micić, J. Ignjatović and M. Ćirić, *Further improvements of determinization methods for fuzzy finite automata*. Fuzzy Sets and Systems, submitted for publication (http://arxiv.org/pdf/1402.6510v3).
19. G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic, Theory and Application*, Prentice-Hall, Englevood Cliffs, NJ, 1995.
20. S. Konstantinidis, N. Santean and S. Yu, *Fuzzification of rational and recognizable sets*, Fundamenta Informaticae **76** (2007), 413–447.
21. O. Kupferman and Y. Lustig, *Lattice automata*, in: Proceedings of VWCAI2007, Lecture Notes in Computer Science **4349** (2007), pp. 199–213.
22. E. T. Lee and L. A. Zadeh, *Note on fuzzy languages*, Information Sciences **1** (1969), 421–434.
23. H. Lei and Y. M. Li, *Minimization of states in automata theory based on finite lattice-ordered monoids*, Information Sciences **177** (2007), 1413–1421.
24. P. Li and Y. M. Li, *Algebraic properties of LA-languages*, Information Sciences **176** (2006), 3232–3255.
25. Y. M. Li, *Finite automata theory with membership values in lattices*, Information Sciences **181** (2011), 1003–1017.
26. Y. M. Li and W. Pedrycz, *Fuzzy finite automata and fuzzy regular expressions with membership values in lattice ordered monoids*. Fuzzy Sets and Systems **156** (2005), 68–92.
27. Y. M. Li and W. Pedrycz, *Minimization of lattice finite automata and its application to the decomposition of lattice languages*, Fuzzy Sets and Systems **158** (2007), 1423–1436.
28. Z. Li, P. Li and Y. M. Li, *The relationships among several types of fuzzy automata*, Information Sciences **176** (2006), 2208–2226.
29. F. Lin, H. Ying, R.D. MacArthur, J. A. Cohn, D. Barth-Jones and L. R. Crane, *Decision making in fuzzy discrete event systems*. Information Sciences **177** (2007), 3749–3763.
30. F. Liu and D. Qiu, *Decentralized supervisory control of fuzzy discrete event systems*. European Journal of Control **3** (2008), 234–243.
31. F. Liu and D. Qiu, *Diagnosability of fuzzy discrete-event systems: A fuzzy approach*. IEEE Transactions on Fuzzy Systems **17 (2)** (2009), 372–384.
32. I. Micić, Z. Jančić, J. Ignjatović and M. Ćirić, *Determinization of fuzzy automata by means of the degrees of language inclusion*. IEEE Transactions on Fuzzy Systems (2015), DOI:

10.1109/ TFUZZ.2015.2404348.
33. J. N. Mordeson and D. S. Malik, *Fuzzy Automata and Languages: Theory and Applications*, Chapman & Hall/CRC, Boca Raton, London, 2002.
34. K. Peeva, *Finite L-fuzzy acceptors, regular L-fuzzy grammars and syntactic pattern recognition*, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **12** (2004), 89–104.
35. K. Peeva, *Finite L-fuzzy machines*, Fuzzy Sets and Systems **141** (2004), 415–437.
36. K. Peeva and Y. Kyosev, *Fuzzy Relational Calculus: Theory, Applications, and Software (with CD-ROM)*, in Series "Advances in Fuzzy Systems – Applications and Theory", Vol 22, World Scientific, 2004.
37. K. Peeva and Z. Zahariev, *Computing behavior of finite fuzzy machines – Algorithm and its application to reduction and minimization*, Information Sciences **178** (2008), 4152–4165.
38. D. W. Qiu, *Automata theory based on completed residuated lattice-valued logic (I)*, Science in China, Ser. F, **44 (6)** (2001), 419–429.
39. D. W. Qiu, *Automata theory based on completed residuated lattice-valued logic (II)*, Science in China, Ser. F, **45 (6)** (2002), 442–452.
40. D. W. Qiu, *Characterizations of fuzzy finite automata*, Fuzzy Sets and Systems **141** (2004), 391–414.
41. D. W. Qiu, *Pumping lemma in automata theory based on complete residuated lattice-valued logic: A note*, Fuzzy Sets and Systems **157** (2006), 2128–2138.
42. D. Qiu and F. Liu, *Fuzzy discrete-event systems under fuzzy observability and a test algorithm*. IEEE Transactions on Fuzzy Systems **17 (3)** (2009) 578–589.
43. E. S. Santos, *Maximin automata*, Information and Control **12** (1968), 367–377.
44. E. S. Santos, *On reduction of maxmin machines*, Journal of Mathematical Analysis and Applications **37** (1972), 677–686.
45. E. S. Santos, *Fuzzy automata and languages*, Information Sciences **10** (1976), 193–197.
46. A. Stamenković and M. Ćirić, *Construction of fuzzy automata from fuzzy regular expressions*, Fuzzy Sets and Systems **199** (2012), 1–27.
47. A. Stamenković, M. Ćirić and J. Ignjatović, *Reduction of fuzzy automata by means of fuzzy quasi-orders*, Information Sciences **275** (2014), 168–198.
48. W. Wechler, *The Concept of Fuzziness in Automata and Language Theory*, Akademie-Verlag, Berlin, 1978.
49. W. G. Wee, *On generalizations of adaptive algorithm and application of the fuzzy sets concept to pattern classification*, Ph.D. Thesis, Purdue University, 1967.
50. W. G. Wee and K. S. Fu, *A formulation of fuzzy automata and its application as a model of learning systems*, IEEE Transactions on Systems, Man and Cybernetics **5** (1969), 215–223.
51. L. H. Wu and D. W. Qiu, *Automata theory based on complete residuated lattice-valued logic: Reduction and minimization*, Fuzzy Sets and Systems **161** (2010), 1635–1656.
52. H. Xing and D. W. Qiu, *Pumping lemma in context-free grammar theory based on complete residuated lattice-valued logic*, Fuzzy Sets and Systems **160** (2009), 1141–1151.
53. H. Xing and D. W. Qiu, *Automata theory based on complete residuated lattice-valued logic: A categorical approach*, Fuzzy Sets and Systems **160** (2009), 2416–2428.
54. H. Xing, D. W. Qiu and F. C. Liu, *Automata theory based on complete residuated lattice-valued logic: Pushdown automata*, Fuzzy Sets and Systems **160** (2009), 1125–1140.
55. H. Xing, D. W. Qiu, F. C. Liu and Z. J. Fan, *Equivalence in automata theory based on complete residuated lattice-valued logic*, Fuzzy Sets and Systems **158** (2007), 1407–1422.
56. H. Xing, Q. Zhang and K. Huang, *Analysis and control of fuzzy discrete event systems using bisimulation equivalence*, Theoretical Computer Science **456** (2012), 100–111.

Aleksandar Stamenković
Faculty of Sciences and Mathematics
Department of Computer Science
Višegradska 33, P. O. Box 224
18000 Niš, Serbia
aca@pmf.ni.ac.rs

Miroslav Ćirić
Faculty of Sciences and Mathematics
Department of Computer Science
Višegradska 33, P. O. Box 224
18000 Niš, Serbia
miroslav.ciric@pmf.edu.rs

Jelena Ignjatović
Faculty of Sciences and Mathematics
Department of Computer Science
Višegradska 33, P. O. Box 224
18000 Niš, Serbia
jelena.ignjatovic@pmf.edu.rs