N-Fold Fuzzy Grammars

MASAHARU MIZUMOTO, JUNICHI TOYODA, AND KOHKICHI TANAKA
Faculty of Engineering Science, Osaka University,
Toyonaka, Osaka, Japan 560

Communicated by John M. Richardson

ABSTRACT

A new form of fuzzy grammar, which is called an n-fold fuzzy grammar, is defined and some of its properties are investigated. The n-fold fuzzy grammars are a generalization of fuzzy grammars defined by Lee and Zadeh, where the grade of application of the rule to be used next is conditioned by the $n(\geqslant 1)$ rules used before in a derivation. The n-fold fuzzy grammars whose rules are of context-free form can be shown to generate context-sensitive languages by setting a threshold appropriately. Fuzzy grammars by Lee and Zadeh with context-free rules, however, cannot generate context-sensitive languages. As to n-fold fuzzy grammars, we mainly focused our attention on n-fold fuzzy grammars with type 3 rules as a preliminary step.

INTRODUCTION

Natural languages such as English have incorrectness and ambiguity syntactically and semantically. It is natural to introduce randomness into the structure of formal languages in order to specify natural languages with ambiguity [1–3]. Another way of extending the concepts of formal languages to those of natural languages is the introduction of fuzziness. The first step to this direction was made by Lee and Zadeh [4], who introduced fuzzy grammars as an extension of ordinary formal grammars [5] by using the concept of fuzzy sets [6].

In this paper we shall discuss fuzzy grammars by Lee and Zadeh, and $n(\geqslant 1)$-fold fuzzy grammars which are a generalization of fuzzy grammars.

Ordinary formal grammars have the property that, after applying a rewriting rule to an intermediate string in a derivation, the next rewriting rule to be used can be chosen arbitrarily. This arbitrariness, however, is not sufficient for describing natural languages with fuzziness. For example, consider the following rewriting rules: (1) $S \to A$ bites B, (2) $A \to$ the dog, (3) $A \to$ the boy, (4) $B \to$ the boy, (5) $B \to$ the dog. Then the sentences generated by these rewriting rules with an initial symbol S are as follows. {the dog bites the boy, the dog bites the dog, the boy bites the boy, the boy bites the dog}. Generally speaking, a sentence "the boy bites the dog" is rather doubtful semantically. We may say that, after applying the rewriting rule (3) to the intermediate string "A bites B", it is not rather proper to apply the rewriting rule (5).

To specify such a condition, we have defined n-fold fuzzy grammars, in which the grade (or the propriety) of the application of the rewriting rule to be used next is conditioned by the $n(\geqslant 1)$ rules used before in a derivation. The grade approaches to unity nearer and nearer as the propriety becomes higher. In the case of applying several rewriting rules, we use the concept of composition of fuzzy relations [6].

We show that the set of all strings whose grades of the generation by fuzzy grammars by Lee and Zadeh [4] with type i $(i = 0, 1, 2, 3)$ rules are greater than a certain threshold, is a type i language and that the set of all strings whose grades by fuzzy grammars with type 2 rules are between two thresholds is not necessary a type 2 language.

n-Fold fuzzy grammars whose rules are of the type 2 form can be shown to generate type 1 languages by setting a threshold appropriately. As to n-fold fuzzy grammars, we focused our attention on n-fold fuzzy grammars with type 3 rules as a preliminary step.

1. FUZZY GRAMMARS

The notion of fuzzy grammars defined by Lee and Zadeh [4] is a natural generalization of the definition of formal grammars [5]. In this section we shall show that the set of all strings whose grades of the generation obtained by fuzzy grammars with type i $(i = 0, 1, 2, 3)$ rules are greater than a certain threshold, is a type i language. But the set of all strings whose grades by fuzzy grammars with type 2 rules are between two thresholds is not necessary a type 2 language. The same holds for fuzzy grammars with type 0 rules.

Definition 1.1. A *fuzzy grammar* (FG for short) is a system

$$\mathrm{FG} = (V_N, V_T, P, S, J, f), \tag{1.1}$$

where

(i) V_N is a nonterminal vocabulary.
(ii) V_T is a terminal vocabulary.
(iii) S is an initial symbol in V_N.

(iv) P is a finite set of productions such as

$$(r)\ u \to v\ f(r), \tag{1.2}$$

where $r \in J$, $u \to v$ is an ordinary rewriting rule with $u \in V_N^* - \{\epsilon\}$, $v \in (V_N \cup V_T)^*$, and $f(r)$ is the grade of the application of the production r, which will be denoted in (vi).

(v) J is a set of (rewriting rule) labels as shown in (iv).

$$J = \{r\}.$$

(vi) f is a membership function such as

$$f{:}J \to [0, 1]. \tag{1.3}$$

f may be called a *fuzzy function* and the value $f(r)$, $r \in J$, is the grade of the application of a production r.[1]

We assume that, to each rewriting rule, there can correspond more than one label, but not conversely.

Next, we shall briefly explain a derivation chain with fuzzy grades (fuzzy derivation chain).

If $(r)u \to vf(r)$ is in P, and α and β are any strings in $(V_N \cup V_T)^*$, then

$$\alpha u\beta \xrightarrow[r]{f(r)} \alpha v\beta, \tag{1.4}$$

and $\alpha v\beta$ is said to be directly derivable from $\alpha u\beta$ with the grade $f(r)$ by the production r. If $\alpha_1, \alpha_2, \ldots, \alpha_m$ are strings in $(V_N \cup V_T)^*$ and

$$\alpha_0 \xrightarrow[r_1]{f(r_1)} \alpha_1, \qquad \alpha_1 \xrightarrow[r_2]{f(r_2)} \alpha_2, \ldots, \alpha_{m-1} \xrightarrow[r_m]{f(r_m)} \alpha_m, \tag{1.5}$$

then α_m is said to be derivable from α_0 by the productions $r_1, r_2, \ldots, r_m$. The expression

$$\alpha_0 \xrightarrow[r_1]{f(r_1)} \alpha_1 \xrightarrow[r_2]{f(r_2)} \alpha_2 \longrightarrow \cdots \xrightarrow[r_m]{f(r_m)} \alpha_m, \tag{1.6}$$

will be referred to as a *fuzzy derivation chain* of length m from α_0 to α_m by the productions $r_1, r_2, \ldots, r_m$.

When $\alpha_0 = S$, $\alpha_m = x (\in V_T^*)$ in Eq. 1.1, i.e.

$$S \xrightarrow[r_1]{f(r_1)} \alpha_1 \xrightarrow[r_2]{f(r_2)} \alpha_2 \longrightarrow \cdots \longrightarrow \alpha_{m-1} \xrightarrow[r_m]{f(r_m)} x, \tag{1.7}$$

S is said to generate a terminal string x by the productions $r_1, r_2, \ldots, r_m$. In general, there are more than one fuzzy derivation chain from S to x.

[1] We often say the label r is the production r for convenience.

Definition 1.2. The grade of the generation of terminal string $x(\in V_T{}^*)$ by a fuzzy grammar FG, which is denoted as $f_{FG}(x)$, is given as follows by using the concept of the composition of fuzzy relations [6] and by the fuzzy derivation chain from S to x of Eq. 1.7. Clearly, $f_{FG}(x)$ is in [0.1].

$$f_{FG}(x) = \max \min [f(r_1), f(r_2), \ldots, f(r_m)], \tag{1.8}$$

where the maximum is taken over all the fuzzy derivation chains from S to x.

We shall define a language generated by a fuzzy grammar FG with the threshold λ, $0 \leqslant \lambda < 1$.

Definition 1.3. Let $\mathrm{FG} = (V_N, V_T, P, S, J, f)$ be a fuzzy grammar and λ a real number $0 \leqslant \lambda < 1$, then a language generated by FG with the threshold λ is defined by

$$\mathrm{L}(\mathrm{FG}, \lambda) = \{x \in V_T{}^* \mid f_{FG}(x) > \lambda\}. \tag{1.9}$$

Moreover, we can also define other languages as follows:

Definition 1.4. For two thresholds λ_1 and λ_2 with $0 \leqslant \lambda_1 < \lambda_2 < 1$, a language $\mathrm{L}(\mathrm{FG}, \lambda_1, \lambda_2)$ is defined by the following:

$$\mathrm{L}(\mathrm{FG}, \lambda_1, \lambda_2) = \{x \in V_T{}^* \mid \lambda_1 < f_{FG}(x) \leqslant \lambda_2\}. \tag{1.10}$$

Besides, for a threshold λ, a language $\mathrm{L}(\mathrm{FG}, =, \lambda)$ is defined as

$$\mathrm{L}(\mathrm{FG}, =, \lambda) = \{x \in V_T{}^* \mid f_{FG}(x) = \lambda\}, \tag{1.11}$$

where λ is $0 \leqslant \lambda \leqslant 1$.

Example 1.1. Let FG be the fuzzy grammar (V_N, V_T, P, S, J, f), where $V_N = \{S, A, B, C, D, E\}$, $V_T = \{a, b\}$, and the productions are

(1)	$S \to ABC$	0.5,	(8)	$B \to bB$	0.8
(2)	$S \to ADC$	0.8,	(9)	$B \to b$	0.9
(3)	$S \to DBC$	0.7,	(10)	$C \to aC$	0.8
(4)	$S \to ABE$	0.6,	(11)	$C \to a$	0.9
(5)	$S \to AEC$	0.6,	(12)	$D \to aDb$	0.8
(6)	$A \to aA$	0.8,	(13)	$D \to ab$	0.8
(7)	$A \to a$	0.9,	(14)	$E \to bEa$	0.8
			(15)	$E \to ba$	0.8.

A string, say, a^2b^2a is obtained by the following derivation.

$$\mathbf{S} \xrightarrow[1]{0.5} \mathbf{A}BC \xrightarrow[6]{0.8} a\mathbf{A}BC \xrightarrow[7]{0.9} a^2\mathbf{B}C \xrightarrow[8]{0.8} a^2b\mathbf{B}C \xrightarrow[9]{0.9} a^2b^2\mathbf{C} \xrightarrow[11]{0.9} a^2b^2a,$$

where the bold symbol in the intermediate string represents the location where the next production was applied.

The grade of the generation of a^2b^2a by this derivation is the minimum value among the values indicated over the arrows, i.e.

$$\min(0.5, 0.8, 0.9, 0.8, 0.9, 0.9) = 0.5.$$

Similarly, for the same string a^2b^2a, the following derivation is also possible,

$$\mathbf{S} \xrightarrow[4]{0.6} \mathbf{A}B E \xrightarrow[6]{0.8} a\mathbf{A}BE \xrightarrow[7]{0.9} a^2\mathbf{B}E \xrightarrow[9]{0.9} a^2b\mathbf{E} \xrightarrow[15]{0.8} a^2b^2a.$$

In this case, we have 0.6. Furthermore, we can show other derivations of a^2b^2a, whose grades are shown to be less than 0.6. Thus we have $f_{\mathrm{FG}}(a^2b^2a) = 0.6$ from Eq. 1.8.

Continuing in this manner, we can see that the languages generated by FG are, for example

(i) $\mathrm{L}(\mathrm{FG}, 0.45) = \{a^i b^j a^k | i, j, k \geqslant 1\}$,
(ii) $\mathrm{L}(\mathrm{FG}, 0.55) = \{a^i b^j a^k | i, j, k \geqslant 1, i \neq j \text{ or } j \neq k\}$,
(iii) $\mathrm{L}(\mathrm{FG}, 0.75) = \{a^{i+j} b^j a^k | i, j, k \geqslant 1\}$,
(iv) $\mathrm{L}(\mathrm{FG}, 0.45, 0.55) = \mathrm{L}(\mathrm{FG}, =, 0.5) = \{a^i b^i a^i | i \geqslant 1\}$.

It is interesting to note that languages (i), (ii), and (iii) given above are all context-free languages, but languages (iv) are context-sensitive languages.

Remark: In this example, the language $\{a^i b^j a^k | i, j, k \geqslant 1\}$ is generated by the rewriting rules whose labels are (1) and (6) ~ (11). And the language $\{a^i b^j a^k | i \neq j \text{ or } j \neq k\}$ is by the rewriting rules of (2) ~ (15).

Definition 1.5. A fuzzy grammar FG in which the rewriting rules are of type i ($i = 0, 1, 2, 3,$) is denoted as i-FG. The language by i-FG with the threshold λ is defined as $\mathrm{L}(i\text{-FG}, \lambda)$.

THEOREM 1.1. *For any* $\lambda (0 \leqslant \lambda < 1)$, *a language* $\mathrm{L}(i\text{-FG}, \lambda)$ *is a type i language in the sense of Chomsky, where* $i = 0, 1, 2, 3$.

Proof: For the $i\text{-FG} = (V_N, V_T, P, S, J, f)$ with the threshold λ $(0 \leqslant \lambda < 1)$, let $J(\lambda)$ be the set of all labels such that $f(r) > \lambda$, where $r \in J$. More precisely, $J(r) = \{r | f(r) > \lambda\}$. Then $\mathrm{L}(i\text{-FG}, \lambda)$ is a language which was obtained from the only rewriting rules corresponding to the labels in $J(\lambda)$. $\mathrm{L}(i\text{-FG}, \lambda)$ is, therefore, a type i language, where $i = 0, 1, 2, 3$.

THEOREM 1.2. *For the i*-FG, *where* $i = 0, 2$, *the languages* $\mathrm{L}(i\text{-FG}, \lambda_1, \lambda_2)$ *and* $\mathrm{L}(i\text{-FG}, =, \lambda)$ *are not always type i languages.*

Proof: The language $\mathrm{L}(i\text{-FG}, \lambda_1, \lambda_2)$ is given by the difference $\mathrm{L}(i\text{-FG}, \lambda_1) - \mathrm{L}(i\text{-FG}, \lambda_2)$. We know that the type i $(i = 0, 2)$ languages are not always closed under the difference. Thus, the languages L(i-FG, λ_1) and L(i-FG, λ_2) are type i languages, so L(i-FG, λ_1, λ_2) is not always a type i language, where $i = 0$, 2. Similarly, L(i-FG, $=$, λ) is given by L(i-FG, $\geqslant$, λ) $-$ L(i-FG, λ), where L(i-FG, $\geqslant$, λ) is defined as $\{x \in V_T^* | f_{FG}(x) \geqslant \lambda\}$. Clearly, L($i$-FG, $\geqslant$, λ) is a type i language, so L(i-FG, $=$, λ) is not always type i language, where $i = 0$, 2.

Note: L(3-FG, λ_1, λ_2) and L(3-FG, $=$, λ) are also type 3 languages, since type 3 languages are closed under the difference. We can not, however, conclude whether L(1-FG, λ_1, λ_2) and L(1-FG, $=$, λ) are type 1 languages or not, because it is not known whether type 1 languages are closed under the difference or not.

2. *N*-FOLD FUZZY GRAMMARS

In this section we shall define an n-fold fuzzy grammar in which the grade of the application of the rewriting rule to be used next is conditioned by the n rules used before in a derivation, where $n \geqslant 1$. And it is shown that n-fold fuzzy grammars with CF (or type 2) rules can generate CS (or type 1) languages.

Definition 2.1. An $n(\geqslant 1)$*-fold fuzzy grammar* (n-FG for short) is a system,

$$n\text{-FG} = (V_N, V_T, P, S, J, \{f_1, f_2, \ldots, f_n\}), \tag{2.1}$$

where V_N, V_T, S, and J have essentially the same meanings as those for the fuzzy grammars denoted in the previous section, and P is a finite set of rules with labels such as

$$(r)\, u \to v, \tag{2.2}$$

where $r \in J$, $u \to v$ is an ordinary rewriting rule with $u \in V_N^* - \{\epsilon\}$ and $v \in (V_N \cup V_T)^*$. $J = \{r\}$ is a set of (rewriting rule) labels. f_i ($i = 0, 1, \ldots, n$) is a (conditional) membership function of a fuzzy set in the label set J and is defined as follows:

(a) In the case of $i = 0$: f_0 is a membership function from J_S to $[0, 1]$, i.e.

$$f_0 : J_S \to [0, 1], \tag{2.3}$$

where J_S is the set of all labels whose rules are initial rules. The value $f_0(r)$ in $[0, 1]$ represents the grade of the application of an initial rule r in J_S.

(b) In the case of $1 \leqslant i \leqslant n$: f_i is a conditional membership function such as

$$f_i(r_1, r_2, \ldots, r_i; r_{i+1}) \in [0, 1], \tag{2.4}$$

and represents the grade of membership of r_{i+1} in J given r_1, r_2, ..., r_i in J. In other words, $f_i(r_1, r_2, \ldots, r_i; r_{i+1})$ is designated as the grade of the application of the rule r_{i+1} after the i rules r_1, r_2, ..., r_i were applied sequentially to the intermediate string in a derivation.

In what follows, we shall call f_i $(i = 0, 1, \ldots, n)$ as an *i-fold fuzzy function.*

We assumed that, to each rule, there may correspond more than one label, but not conversely.

Remark: For all r in J_S, all i $(i = 1, 2, \ldots, n)$, and all $r_1, r_2, \ldots, r_i$ in J, let

$$f_0(r) = f(r),$$

$$f_i(r_1, r_2, \ldots, r_i; r_{i+1}) = f(r_{i+1}).$$

Then n-fold fuzzy grammar becomes a fuzzy grammar denoted in the previous section. Thus we may call a fuzzy grammar as a *0-fold fuzzy grammar.*

Now, we shall explain how to use i-fold fuzzy functions f_i, $i = 0, 1, \ldots, n$, in a derivation.

The expression

$$S \xrightarrow[r_1]{} \alpha_1 \xrightarrow[r_2]{} \alpha_2 \xrightarrow[r_3]{} \cdots \xrightarrow[r_m]{} \alpha_m, \tag{2.5}$$

will be referred to as a derivation chain of length m by the rules $r_1, r_2, \ldots, r_m$ from S to α_m.

When the length m of this derivation chain is $m \leqslant n$, the fuzzy functions $f_0, f_1, \ldots, f_{m-1}$ are employed as follows:

Let

$$f_0(r_1) = \mu_1, f_1(r_1; r_2) = \mu_2, f_2(r_1, r_2; r_3) = \mu_3, \ldots,$$

and

$$f_{m-1}(r_1, r_2, \ldots, r_{m-1}; r_m) = \mu_m{}^2,$$

then we put each fuzzy grade $\mu_1, \mu_2, \ldots, \mu_m \in [0, 1]$ over the arrow by the following.

$$S \xrightarrow[r_1]{\mu_1} \alpha_1 \xrightarrow[r_2]{\mu_2} \alpha_2 \longrightarrow \cdots \xrightarrow[r_{m-1}]{\mu_{m-1}} \alpha_{m-1} \xrightarrow[r_m]{\mu_m} \alpha_m. \tag{2.6}$$

Moreover, when the length m is $m > n$, we let $m = n + j$, $j \geqslant 1$. Then, in general, after the n rules $r_j, r_{j+1}, \ldots, r_{n+j-1}$, where $j \geqslant 1$, were applied sequentially to the intermediate string, the grade of the application of the rule r_{n+j} is characterized by an n-fold fuzzy function f_n. Let $f_n(r_j, r_{j+1}, \ldots, r_{n+j-1}; r_{n+j}) = \mu_{n+j}$, $j \geqslant 1$, then the grades $\mu_{n+1}, \mu_{n+2}, \ldots, \mu_{n+j}$ are expressed as follows, where $\mu_1, \mu_2, \ldots, \mu_n$ are dependent on the fuzzy functions $f_0, f_1, \ldots, f_{n-1}$ as mentioned before.

$$S \xrightarrow[r_1]{\mu_1} \alpha_1 \xrightarrow[r_2]{\mu_2} \cdots \xrightarrow[r_n]{\mu_n} \alpha_n \xrightarrow[r_{n+1}]{\mu_{n+1}} \alpha_{n+1} \longrightarrow \cdots \xrightarrow[r_{n+j}]{\mu_{n+j}} \alpha_{n+j}.$$

---

[2] The label r_1 in i-fold fuzzy function f_i $(i = 0, 1, \ldots, n-1)$ is the label whose rule is an initial rule.

We shall call this derivation chain with fuzzy grades as *fuzzy derivation chain.*

We shall next explain a fuzzy language characterized by an n-fold fuzzy grammar n-FG.

Let Eq. 2.8 be a fuzzy derivation chain from an initial symbol S to terminal string x in $V_T{}^*$, that is,

$$S \xrightarrow[r_1]{\mu_1} \alpha_1 \xrightarrow[r_2]{\mu_2} \alpha_2 \longrightarrow \cdots \xrightarrow[r_k]{\mu_k} \alpha_k\,(=x). \tag{2.8}$$

Then the grade of the generation of x by this fuzzy derivation chain is defined as

$$\min\,[\mu_1, \mu_2, \ldots, \mu_k]. \tag{2.9}$$

By using the concept of composition of fuzzy relations, the grade of the generation of x in $V_T{}^*$ by n-FG is given as follows.

$$\mu_{n\text{-FG}}(x) = \max\min\,[\mu_1, \mu_2, \ldots, \mu_k], \tag{2.10}$$

where the maximum is taken over all the fuzzy derivation chains from S to x.

Definition 2.2. A *fuzzy language* by n-FG is a fuzzy set in $V_T{}^*$ characterized by the membership function $\mu_{n\text{-FG}}(x)$ as defined in Eq. 2.10 and may be called an *n-fold fuzzy language* which is denoted as L(n-FG) which is represented as $\{(x, \mu_{n\text{-FG}}(x))\}$ for $x \in V_T{}^*$ [4].

Especially, we call an n-fold fuzzy grammar with the grade of the generation of the terminal string under the operation "maxmin" as in Eq. 2.10 as an *n-fold pessimistic fuzzy grammar* (=n-PFG), and an n-fold fuzzy grammar under the operation "minmax" which is denoted in Eq. 2.11 as an *n-fold optimistic fuzzy grammar* (=n-OFG) [11].

$$\mu_{n\text{-OFG}}(x) = \min\max\,[\mu_1, \mu_2, \ldots, \mu_k]. \tag{2.11}$$

A fuzzy language characterized by $\mu_{n\text{-OFG}}$ is denoted as L(n-OFG).

In this paper, unless stated especially, by an "n-fold fuzzy grammar" we shall mean an n-fold pessimistic fuzzy grammar.

Example 2.1. Consider the following 2-fold fuzzy grammar,

$$2\text{-FG} = (V_N, V_T, P, S, J, \{f_0, f_1, f_2\}),$$

where $V_N = \{S, A, B\}$, $V_T = \{a, b, c\}$, P consists of the following.

(1) $S \to AB$, (4) $B \to cB$,

(2) $A \to aAb$, (5) $B \to c$.

(3) $A \to ab$,

The 0, 1, 2-fold fuzzy functions are

$$
\begin{array}{ll}
f_0(1)=1, & \\
f_1(1;3)=0.9, & f_1(1;2)=0.8, \\
f_2(1,3;5)=0.9, & f_2(1,2;2)=0.8, \\
f_2(2,2;4)=0.7, & f_2(2,4;4)=0.7, \\
f_2(4,4;2)=0.7, & f_2(4,2;2)=0.7, \\
f_2(4,4;3)=0.6, & f_2(4,3;5)=0.6.
\end{array}
$$

and all other f_1 and f_2 are 0.5.

Now, we shall obtain the grades of the generation of the terminal strings by this 2-FG. A string, say, $a^3b^3c^3$ is obtained by the following fuzzy derivation chain.

$$
\mathbf{S} \xrightarrow[1]{1} \mathbf{A}B \xrightarrow[2]{0.8} a\mathbf{A}b B \xrightarrow[2]{0.8} a^2 Ab^2 \mathbf{B} \xrightarrow[4]{0.7} a^2 Ab^2 c\mathbf{B}
$$

$$
\xrightarrow[4]{0.7} a^2 \mathbf{A}b^2 c^2 B \xrightarrow[3]{0.6} a^3 b^3 c^2 \mathbf{B} \xrightarrow[5]{0.6} a^3 b^3 c^3.
$$

The grade of the generation of $a^3b^3c^3$ by this derivation is given from Eq. 2.9 as follows.

$$
\min[1, 0.8, 0.8, 0.7, 0.7, 0.6, 0{,}6] = 0.6.
$$

The 0, 1, 2-fold fuzzy functions used sequentially in this derivation are

$$
\begin{array}{lll}
f_0(1)=1, & f_1(1;2)=0.8, & f_2(1,2;2)=0.8, \\
f_2(2,2;4)=0.7, & f_2(2,4;4)=0.7, & f_2(4,4;3)=0.6, \\
f_2(4,3;5)=0.6. & &
\end{array}
$$

Similarly, for the same string $a^3b^3c^3$, the following derivation is also possible.

$$
\mathbf{S} \xrightarrow[1]{1} A\mathbf{B} \xrightarrow[4]{0.5} \mathbf{A}cB \xrightarrow[2]{0.5} aAbc\mathbf{B} \xrightarrow[4]{0.5} a\mathbf{A}bc^2 B
$$

$$
\xrightarrow[4]{0.7} a^2 \mathbf{A}b^2 c^2 B \xrightarrow[3]{0.6} a^3 b^3 c^2 \mathbf{B} \xrightarrow[5]{0.6} a^3 b^3 c^3.
$$

In this case, we have

$$
\min[1, 0.5, 0.5, 0.5, 0.7, 0.6, 0.6] = 0.5.
$$

Furthermore, we can also show the different fuzzy derivation chains of $a^3b^3c^3$, whose all grades are shown to be 0.5.

Hence, the grade of the generation of $a^3b^3c^3$ by this 2-FG is given as the maximum value among the grades for all the fuzzy derivation chains of $a^3b^3c^3$ (see Eq. 2.10). Thus we have $\mu_{2\text{-FG}}(a^3b^3c^3) = 0.6$.

Continuing in this manner, we can see that the fuzzy language characterized by 2-FG is

$$\begin{aligned} \mathrm{L}(2\text{-FG}) &= \{(x, \mu_{2\text{-FG}}(x))\} \\ &= \{(abc, 0.9)\} \cup \{(a^{2n+1}b^{2n+1}c^{2n+1}, 0.6) \mid n \geqslant 1\} \\ &\quad \cup \{(a^i b^i c^j, 0.5) \mid (i,j) \neq (2n-1, 2n-1), i,j,n \geqslant 1\}. \end{aligned}$$

Let $\mathrm{L}(2\text{-FG}, \lambda) = \{x \in V_T^* \mid \mu_{2\text{-FG}}(x) > \lambda\}$, where $0 \leqslant \lambda < 1$, and let $\lambda = 0.85$, 0.55, 0.45. Then

$$\begin{aligned} &\mathrm{L}(2\text{-FG}, 0.85) = \{abc\}, \\ &\mathrm{L}(2\text{-FG}, 0.55) = \{a^{2n-1}b^{2n-1}c^{2n-1} \mid n \geqslant 1\}, \\ &\mathrm{L}(2{:}\mathrm{FG}, 0.45) = \{a^i b^i c^j \mid i,j \geqslant 1\}. \end{aligned}$$

In the case of $n = 1$ in n-FG, 1-fold fuzzy function f_1 of

$$1\text{-FG} = (V_N, V_T, P, S, J, \{f_0, f_1\})$$

can be represented by the m fuzzy vectors whose dimension is m, where $m = \#(J)$. That is, let $J = \{r_1, r_2, \ldots, r_m\}$, then for each $(r_i)u_i \to v_i$ in P,

$$(r_i)\, u_i \to v_i\ (f_{r_i r_1}, f_{r_i r_2}, \ldots, f_{r_i r_m}), \tag{2.12}$$

where $f_{r_i r_j} = f_1(r_i; r_j)$ and $i, j = 1, 2, \ldots, m$.

Example 2.2. Let 1-FG be $(V_N, V_T, P, S, J, \{f_0, f_1\})$, where $V_N = \{S, A, B, C\}$, $V_T = \{a, b, c\}$, $f_0(1) = 0.9$, and the rules with fuzzy vectors are

		1	2	3	4	5	6	7	8	9	10	
(1) $S \to ABC$	(		0.7			0.8			0.9			)
(2) $A \to aA$	(			0.7								)
(3) $B \to bB$	(				0.7							)
(4) $C \to cC$	(		0.7						0.7			)
(5) $A \to aAa$	(						0.8					)
(6) $B \to bBb$	(							0.8				)
(7) $C \to cCc$	(					0.8			0.8			)
(8) $A \to a$	(									0.9		)
(9) $B \to b$	(										0.9	)
(10) $C \to c$	(											).

We assumed that the values of the blank portions of the fuzzy vectors are in the interval (0,0.65].

A string, say, $a^3b^3c^3$ is obtained by the following fuzzy derivation chain.

$$\mathbf{S} \xrightarrow[1]{0.9} \mathbf{A}BC \xrightarrow[2]{0.7} aA\mathbf{B}C \xrightarrow[3]{0.7} aAbBC \xrightarrow[4]{0.7} a\mathbf{A}bBcC \xrightarrow[2]{0.7} a^2 Ab\mathbf{B}cC$$

$$\xrightarrow[3]{0.7} a^2 Ab^2 Bc\mathbf{C} \xrightarrow[4]{0.7} a^2 \mathbf{A}b^2 Bc^2 C \xrightarrow[6]{0.7} a^3 b^2 \mathbf{B}c^2 C \xrightarrow[9]{0.9} a^3 b^3 c^2 \mathbf{C}$$

$$\xrightarrow[10]{0.9} a^3 b^3 c^3.$$

The grade of the generation of $a^3b^3c^3$ by this derivation is from Eq. 2.9 as follows.

$$\min [0.9, 0.7, 0.7, \ldots, 0.7, 0.9, 0.9] = 0.7.$$

Similarly, other derivation chains of $a^3b^3c^3$ are considered.

$$\mathbf{S} \xrightarrow[1]{0.9} \mathbf{A}BC \xrightarrow[5]{0.8} aAa\mathbf{B}C \xrightarrow[6]{0.8} aAabBb\mathbf{C} \xrightarrow[7]{0.8} a\mathbf{A}abBbcCc$$

$$\xrightarrow[8]{0.8} a^3 b\mathbf{B}bcCc \xrightarrow[9]{0.9} a^3 b^3 c\mathbf{C}c \xrightarrow[10]{0.9} a^3 b^3 c^3.$$

In the above case, we have 0.8. Furthermore, we can also show the different derivations of $a^3b^3c^3$, the grades of which are all less than or equal 0.65. Thus, $f_{1\text{-FG}}(a^3b^3c^3) = 0.8$.

Continuing in this manner, we can see that the languages with the thresholds generated by this 1-FG are, for example,

$$\begin{aligned} L(1\text{-FG}, 0.95) &= \phi, \\ L(1\text{-FG}, 0.85) &= \{abc\}, \\ L(1\text{-FG}, 0.75) &= \{a^{2n-1} b^{2n-1} c^{2n-1} \mid n \geqslant 1\}, \\ L(1\text{-FG}, 0.65) &= \{a^n b^n c^n \mid n \geqslant 1\}, \\ L(1\text{-FG}, 0) &= \{a^p b^q c^r \mid p, q, r \geqslant 1\}, \\ L(1\text{-FG}, 0.65, 0.75) &= \{a^{2n} b^{2n} c^{2n} \mid n \geqslant 1\}, \\ L(1\text{-FG}, =, 0.8) &= \{a^{2n+1} b^{2n+1} c^{2n+1} \mid n \geqslant 1\}, \end{aligned}$$

where the language L(1-FG, 0.65, 0.75) is defined as

$$\{x \in V_T^* \mid 0.65 < \mu_{1\text{-FG}}(x) \leqslant 0.75\}$$

and the language L(1-FG, =, 0.8) is as $\{x \in V_T^* \mid \mu_{1\text{-FG}}(x) = 0.8\}$ (see Definition 1.4).

It is interesting to note that, as shown in Examples 2.1 and 2.2, the languages L(n-FG, λ) by $n(\geqslant 1)$-fold fuzzy grammars n-FG with the rules of CF form can be CS languages. But L(FG,λ) by fuzzy grammars FG (or 0-fold fuzzy grammars) with CF rules denoted in the previous section are CF languages.

3. N-FOLD TYPE 3 FUZZY GRAMMARS

In this section we discuss n-fold fuzzy grammars with type 3 rules only. It is shown that fuzzy languages defined by $n(\geqslant 2)$-fold type 3 fuzzy grammars can be characterized by $(n-1)$-fold type 3 fuzzy grammars and vice versa, thus n-fold type 3 fuzzy grammars can be transformed into 1-fold type 3 fuzzy grammars. Moreover, we show that 1-fold type 3 fuzzy grammars can be transformed into 0-fold type 3 fuzzy grammars (or fuzzy automata) and vice versa.

Definition 3.1. An *n-fold type* 3 *fuzzy grammar* (abbreviated n-FG) is an n-fold fuzzy grammar $(V_N, V_T, P, S, J, \{f_0, f_1, \ldots, f_n\})$ in which each rule in P is of the form:

$$(r)A \to aB \qquad \text{or} \qquad (r)A \to a,$$

where $r \in J$, A, $B \in V_N$, and $a \in V_T$.

Similarly, we can define an *n-fold type* 3 *optimistic fuzzy grammar* (n-OFG for short).

Now, we shall prepare for the following notations in order to put restrictions on the domains of 0, 1, ..., n-fold fuzzy functions without loss of generality.

Let J_{AB} be a set of all the labels such that the nonterminal symbols of the left and the right hand sides of the nonterminal rule in P are A and B ($\in V_N$), respectively. Also, let J_A be a set of all the labels such that the left-hand side of the rule in P is $A(\in V_N)$. Moreover, for nonempty i label sets

$$J_{A_0A_1}, J_{A_1A_2}, \ldots, J_{A_{i-1}A_i},$$

let us define the set of label strings of length i as follows.

$$\begin{aligned} J_{A_0A_1\ldots A_i} &= J_{A_0A_1}J_{A_1A_2}\ldots J_{A_{i-1}A_i} \\ &= \{r_1 r_2 \ldots r_i | r_k \in J_{A_{k-1}A_k}, k = 1, 2, \ldots, i\}, \end{aligned} \tag{3.1}$$

which shows that, after the rule r_k ($k = 1, 2, \ldots, i-1$) was used in a derivation, the next rule r_{k+1} is applicable.

Now, we shall define the i-fold fuzzy function f_i, $i = 0, 1, \ldots, n$, using $J_{A_0A_1\ldots A_i}$ defined above.

(i) As to the 0-fold fuzzy function f_0, let $A_i = S$ in J_{A_i} and define

$$f_0(r) \in [0, 1], \tag{3.2}$$

for each $r \in J_S$.

(ii) As to the i-fold fuzzy function f_i, $i = 1, 2, \ldots, n-1$, let $A_0 = S$ in $J_{A_0A_1\ldots A_i}$ and define

$$f_i(r_1, r_2, \ldots, r_i; r_{i+1}) \in [0, 1], \tag{3.3}$$

for each $r_1 r_2 \ldots r_i \in J_{SA_1\ldots A_i}$ and $r_{i+1} \in J_{A_i}$.

This is carried out for all the nonempty sets $J_{SA_0'\ldots A_i'}$ and $J_{A_i'}$.

(iii) As to the n-fold fuzzy function f_n, define

$$f_n(r_1, r_2, \ldots, r_n; r_{n+1}) \in [0, 1], \tag{3.4}$$

for each $r_1 r_2 \ldots r_n \in J_{A_0A_1\ldots A_n}$ and $r_{n+1} \in J_{A_n}$.

This is carried out for all the nonempty sets $J_{A_0'A_1'\ldots A_n'}$ and $J_{A_n'}$.

Now, we shall show that $n(\geqslant 1)$-fold type 3 fuzzy grammars can be transformed into $(n+1)$-fold type 3 fuzzy grammars and vice versa.

THEOREM 3.1. *For an* n-FG, $n \geqslant 1$, *there exists an* $(n+1)$-FG *such that*

$$\mathrm{L}((n+1)\text{-FG}) = \mathrm{L}(n\text{-FG}). \tag{3.5}$$

Proof: For n-FG $= (V_N, V_T, P, S, J, \{f_0, f_1, \ldots, f_n\})$, let $(n+1)$-FG be $(V_N, V_T, P, S, J, \{h_0, h_1, \ldots, h_n, h_{n+1}\})$, where i-fold fuzzy functions h_i, $i =$ $i = 0, 1, \ldots, n+1$, are as follows.

(a) The case of $0 \leqslant i \leqslant n$: h_i is

$$h_i = f_i, \qquad i = 0, 1, \ldots, n.$$

(b) The case of $i = n+1 : h_{n+1}$ is given by the following.

Let n-fold fuzzy function f_n of n-FG be $f_n(r_1, r_2, \ldots, r_n; r_{n+1})$ and its label string $r_1 r_2 \ldots r_n$ be an element of $J_{A_0A_1\ldots A_n}$ (see Eq. 3.1). Still more, let $J(A_0)$ be the set of labels such that the nonterminal symbol of the right-hand side of the rule in P is $A_0 \in V_N$. Then, h_{n+1} of $(n+1)$-FG is as follows.

Let

$$h_{n+1}(r, r_1, \ldots, r_n; r_{n+1}) = f_n(r_1, \ldots, r_n; r_{n+1}),$$

for every $r \in J(A_0)$.

THEOREM 3.2. *For* $n(\geqslant 2)$-FG, *there exists an* $(n-1)$-FG *such that*

$$\mathrm{L}((n-1)\text{-FG}) = \mathrm{L}(n\text{-FG}). \tag{3.6}$$

Proof: For n-FG $= (V_N, V_T, P, S, J, \{f_0, f_1, \ldots, f_n\})$, let $(n-1)$-FG be $(V_N', V_T, P', S', J', \{h_0, h_1, \ldots, h_{n-1}\})$ where $V_N' = \{\langle 0\rangle\} \cup \{\langle r\rangle | r \in J\}$ and $S' = \langle 0\rangle$. P' is obtained by the following.

(1) For each initial rule in P, that is,

$$(r)\, S \to aA \qquad \text{and} \qquad (r)\, S \to a,$$

let us construct new initial rules in P' as follows.

$$(0,r)\langle 0\rangle \to a\langle r\rangle,$$

and

$$(0,r)\langle 0\rangle \to a.$$

(2) For two nonterminal rules in P such that the nonterminal symbol of the right-hand side of the one rule is coincident with that of the left hand side of the other rule, that is, for the following two rules

$$(r_1)\, A_1 \to a\bar{\mathbf{A}}_2, \qquad (r_2)\, \bar{\mathbf{A}}_2 \to a'\, A_3,$$

define a new nonterminal rule such as

$$(r_1, r_2)\langle r_1\rangle \to a'\langle r_2\rangle.$$

(3) For a nonterminal rule and a terminal rule in P such that the nonterminal symbol of the right hand of the nonterminal rule is coincident with that of the left hand of the terminal rule, that is, for the following two rules

$$(r_1)\, A_1 \to a_1\, \bar{\mathbf{A}}_2, \qquad (r_2)\, \bar{\mathbf{A}}_2 \to a_2,$$

let us construct a new terminal rule such as

$$(r_1, r_2)\langle r_1\rangle \to a_2.$$

Let P' be the set of new rules which were obtained in (1), (2) and (3), and let J' be the set of labels corresponding to these rules.

The i-fold fuzzy function h_i $(i = 0, 1, \ldots, n-1)$ of (n–1)-FG is given as follows.

(a) 0-fold fuzzy function h_0 is

$$h_0((0,r)) = f_0(r), \qquad r \in J_S.$$

(b) i-fold fuzzy function h_i, $i = 1, 2, \ldots, n-2$, is

$$h_i((0,r_1),(r_1,r_2),\ldots,(r_{i-1},r_i);(r_i,r_{i+1})) = f_i(r_1,\ldots,r_i;r_{i+1}).$$

(c) $(n-1)$-fold fuzzy function h_{n-1} is

$$h_{n-1}((0,r_1),\ldots,(r_{n-2},r_{n-1});(r_{n-1},r_n)) = f_{n-1}(r_1,r_2,\ldots,r_{n-1};r_n),$$

and

$$h_{n-1}((r_1,r_2),\ldots,(r_{n-1},r_n);(r_n,r_{n-1})) = f_n(r_1,r_2,\ldots,r_n;r_{n+1}).$$

From the above two theorems 3.1 and 3.2, we can transform $n(\geqslant 2)$ – FG into 1-FG and, conversely, 1-FG into $n(\geqslant 2)$ – FG.

Next, we shall show that 1-FG can be transformed into 0-FG (or fuzzy automaton) and also 0-FG can be transformed into 1-FG.

We now briefly review the concept of fuzzy automata [7,8,9,10]. A fuzzy automaton was proposed by Wee and Fu [7] as a model of pattern recognition and automatic control systems.

Definition 3.2. A *fuzzy automaton A* over the alphabet Σ is a system

$$A = (S, s_1, \{F(a) | a \in \Sigma\}, G), \tag{3.7}$$

where

(i) $S = \{s_1, s_2, \ldots, s_n\}$ is a nonempty finite set of internal states.
(ii) s_1 is the initial state in S.
(iii) G is a subset of S (the set of final states).
(iv) $F(a)$ is a fuzzy matrix of order n (the *fuzzy transition matrix* of A) such that $F(a) = [f_A(s_i, a, s_j)]$, where $a \in \Sigma$, s_i, $s_j \in S$, $1 \leqslant i$, $j \leqslant n$, and $n = \#(S)$.

The function $f_A(s_i, a, s_j)$ is a membership function of a fuzzy set in $S \times \Sigma \times S$; i.e. $f_A : S \times \Sigma \times S \to [0, 1]$. f_A may be called the *fuzzy transition function* and $f_A(s_i, a, s_j)$ is designated as the grade of transition from state s_i to state s_j when the input is a.

The grade of transition for an input string of length m is defined by the composition of fuzzy relations.

Definition 3.3. For an input string $x = a_1 a_2 \ldots a_m \in \Sigma^*$, and s, $t \in S$,

$$f_A(s, x, t) = \max_{q_1, q_2, \ldots, q_{m-1} \in S} \min \, [f_A(s, a_1, q_1), f_A(q_1, a_2, q_2), \ldots, f_A(q_{m-1}, a_m, t)]. \tag{3.8}$$

Definition 3.4. For ϵ, x, $y \in \Sigma^*$,[3] and s, $t \in s$,

$$f_A(s, \epsilon, t) = \begin{cases} 1 \ldots s = t \\ 0 \ldots s \neq t, \end{cases} \tag{3.9}$$

$$f_A(s, xy, t) = \max_{g \in S} \min \, [f_A(s, x, q), f_A(q, y, t)]. \tag{3.10}$$

Definition 3.5. For a fuzzy automaton $A = (S, s_1, \{F(a) | a \in \Sigma\}, G)$, let $\mathrm{L}(A)$ be the fuzzy set in Σ^* which is characterized by the following membership function μ_A:

$$\mu_A(x) = \max_{s_f \in G} f_A(s_1, x, s_f), \tag{3.11}$$

for $x \in \Sigma^*$.

THEOREM 3.3. *Given an* 1-FG, *there exists a fuzzy automaton A such that*

$$\mathrm{L}(A) = \mathrm{L}(1\text{-FG}), \tag{3.12}$$

and vice versa.

---

[3] ϵ is an empty string in Σ^* such that $x\epsilon = \epsilon x = x$ for $x \in \Sigma^*$.

Proof. ($\Rightarrow$) Let 1-FG $= (V_N, V_T, P, \sigma, J, \{f_0, f_1\})$, then a fuzzy automaton $A = (S, s_1, \{F(a) | a \in V_T\}, G)$ is defined as follows.

The set of states is $S = \{\langle 0 \rangle\} \cup \{\langle r \rangle | r \in J\}$, the initial state s_1 is $\langle 0 \rangle$, and the set of final states G is $\{\langle r \rangle | r \in J_{tr}\}$, where J_{tr} is a set of all labels whose rules in 1-FG are terminal rules, i.e.

$$J_{tr} = \{r | (r)\, A \to a\}.$$

Before we obtain the fuzzy transition matrices $F(a)$ with $a \in V_T$, let us introduce a label set J^a. For each $a \in V_T$, define J^a as the set of all labels such that the terminal symbol which appears on the right-hand side of the rule is $a(\in V_T)$. More precisely, we define

$$J^a = \{r | (r)\, A \to \mathbf{a}B\} \cup \{r | (r)\, A \to \mathbf{a}\},$$

for each $a \in V_T$. Clearly we have that, for $a, b \in V_T$

(i) $$J^a \cap J^b = \phi \ldots \text{if } a \neq b,$$

(ii) $$\bigcup_{a \in V_T} J^a = J.$$

Now, we shall obtain the fuzzy transition matrices $F(a)$, $a \in V_T$, of a fuzzy automaton A by the following.

Let $F(a) = [f_A(\langle r \rangle, a, \langle p \rangle)]$, where $\langle r \rangle, \langle p \rangle \in S$, and $r, p \in J \cup \{0\}$.

Four cases arise:

(1) For each rule of the form (r) $A \to aB$ with 1-fold fuzzy function $f_1(r;p)$, where $a \in V_T$ is given, A, $B \in V_N$ are arbitrary, and $p \in J_B$, let

$$f_A(\langle r \rangle, a, \langle p \rangle) = \begin{cases} f_1(r;p) & \text{if } p \in J_B \cap J^a, \\ 0 & \text{otherwise.} \end{cases}$$

(2) For each terminal rule $(r)\, A \to a$, let

$$f_A(\langle r \rangle, a, \langle p \rangle) = 0,$$

for all $p \in J$.

(3) For each 0-fold fuzzy function $f_0(p)$, $p \in J_\sigma$, let $\langle r \rangle = \langle 0 \rangle$ and

$$f_A(\langle 0 \rangle, a, \langle p \rangle) = \begin{cases} f_0(p) & \text{if } p \in J_\sigma \cap J^a, \\ 0 & \text{otherwise.} \end{cases}$$

(4) When $\langle p \rangle = \langle 0 \rangle$, let

$$f_A(\langle r \rangle, a, \langle 0 \rangle) = 0,$$

for all $r \in J$.

($\Leftarrow$) For a fuzzy automaton $A=(S,s_1,\{F(a_k)|a_k \in V_T\},G)$, let 1-FG be $(V_N,V_T,P,\sigma,J,\{f_0,f_1\})$, where $V_N=\{\langle s_i\rangle | s_i \in S\}$, and $\sigma=\langle s_i\rangle$. The rules of 1-FG are given as follows. To the element $f_A(s_i,a_k,s_j)$ of the fuzzy transition matrix $F(a_k)$, corresponds the rule such as $\langle s_i\rangle \to a_k\langle s_j\rangle$ where $1 \leqslant i,j \leqslant n$, $1 \leqslant k \leqslant h$, $n=\#(S)$, and $h=\#(V_T)$. Then the number of the corresponding rules, that is, the number of their labels is n^2h.

Finally, the terminal rules are given as follows.

For each rule $\langle s_i\rangle \to a_k\langle s_j\rangle$ obtained above, if $s_j \in G$, that is, s_j is a final state, then we give a terminal rule as $\langle s_i\rangle \to a_k$. Thus, the number of labels of the terminal rules is hnq, where $q=\#(G)$.

Hence, the total number of labels, i.e. $\#(J)$ is hn^2+hnq ($=t$). We can appropriately attach the labels to the rules obtained above without overlapping. In this paper, the label r of the rule whose form is $(r)\ \langle s_i\rangle \to a_k\langle s_j\rangle$ is in $\{1,2,\ldots,hn^2\}$, and the label r of the terminal rule $(r)\ \langle s_i\rangle \to a_k$ is in $\{hn^2+1,\ldots,t\}$.

Next, we shall obtain 0, 1-fold fuzzy function f_0 and f_1.

[I] 0-fold fuzzy function f_0 is as follows.

(i) If the initial rule is of the form $(r)\ \langle s_1\rangle \to a\langle s\rangle$ where $r \in J_{\langle s_1\rangle}$, and s_1 is an initial state of fuzzy automaton A, then

$$f_0(r)=f_A(s_1,a,s).$$

(ii) If the initial rule is of the form $(r)\langle s_1\rangle \to a$, and let this rule $\langle s_1\rangle \to a$ be obtained from the rule $\langle s_1\rangle \to a\langle s_f\rangle$ for some $s_f \in G$, then

$$f_0(r)=f_A(s_1,a,s_f).$$

[II] 1-fold fuzzy function $f_1(r;p)$ is given as follows, where rth rule is of the form $(r)\ \langle s_1\rangle \to a_k\langle s_j\rangle$, $1 \leqslant r \leqslant hn^2$, and $1 \leqslant p \leqslant t$.

(i) In the case of $1 \leqslant p \leqslant hn^2$:

If the pth rule is $(p)\ \langle s\rangle \to a\langle s'\rangle$, then

$$f_1(r;p)=\begin{cases} f_A(s,a,s') & \text{if } s_j=s,\\ 0 & \text{if } s_j \neq s.\end{cases}$$

(ii) In the case of $hn^2+1 \leqslant p \leqslant t$:

Let the terminal rule $(p)\ \langle s\rangle \to a$ be obtained from the rule $\langle s\rangle \to a\langle s_f\rangle$ for some $s_f \in G$, then

$$f_1(r;p)=\begin{cases} f_A(s,a,s_f) & \text{if } s_j=s,\\ 0 & \text{if } s_j \neq s.\end{cases}$$

Example 3.1. Let 1-FG $= (V_N, V_T, P, \sigma, J, \{f_0, f_1\})$, where $V_N = \{\sigma, A\}$, $V_T = \{a, b\}$, $f_0(1) = 0.9$, $f_0(2) = 0.6$, and the rules with fuzzy vector are

		1	2	3	4	5
(1)	$\sigma \to aA$	(0	0	0.2	0.45	0.1)
(2)	$\sigma \to b\sigma$	(0.5	0.8	0	0	0)
(3)	$A \to aA$	(0	0	0.3	0.35	0.7)
(4)	$A \to a\sigma$	(0.4	0.95	0	0	0)
(5)	$A \to b$	(1	1	1	1	1).

Then we can construct a fuzzy automaton $A = (S, \langle 0\rangle, \{F(a), F(b)\}, G)$, where $S = \{\langle 0\rangle, \langle 1\rangle, \langle 2\rangle, \langle 3\rangle, \langle 4\rangle, \langle 5\rangle\}, G = \{\langle 5\rangle\}$, and the fuzzy transition matrices $F(a)$ and $F(b)$ are given as follows:

$$F(a) = \begin{array}{c} \\ \langle 0\rangle \\ \langle 1\rangle \\ \langle 2\rangle \\ \langle 3\rangle \\ \langle 4\rangle \\ \langle 5\rangle \end{array} \begin{array}{c} \begin{array}{cccccc} \langle 0\rangle & \langle 1\rangle & \langle 2\rangle & \langle 3\rangle & \langle 4\rangle & \langle 5\rangle \end{array} \\ \left[\begin{array}{cccccc} & 0.9 & & & & \\ & & & 0.2 & 0.45 & \\ & 0.5 & & & & \\ & & & 0.3 & 0.35 & \\ & 0.4 & & & & \\ & & & & & \end{array}\right] \end{array},$$

$$F(b) = \begin{array}{c} \\ \langle 0\rangle \\ \langle 1\rangle \\ \langle 2\rangle \\ \langle 3\rangle \\ \langle 4\rangle \\ \langle 5\rangle \end{array} \begin{array}{c} \begin{array}{cccccc} \langle 0\rangle & \langle 1\rangle & \langle 2\rangle & \langle 3\rangle & \langle 4\rangle & \langle 5\rangle \end{array} \\ \left[\begin{array}{cccccc} & & 0.6 & & & \\ & & & & & 0.1 \\ & & 0.8 & & & \\ & & & & & 0.7 \\ & & 0.95 & & & \\ & & & & & \end{array}\right] \end{array}.$$

Note that the values of the blank portions of $F(a)$ and $F(b)$ are equal to 0.

Example 3.2. Let $A = (S, s_1, \{F(a) | a \in V_T\}, G)$ be a fuzzy automaton, where $S = \{s_1, s_2\}$, $V_T = \{a, b\}$, $G = \{s_1, s_2\}$, and the fuzzy transition matrices are

$$F(a) = \begin{array}{c} \\ s_1 \\ s_2 \end{array} \begin{array}{c} \begin{array}{cc} s_1 & s_2 \end{array} \\ \begin{bmatrix} 0.5 & 0.4 \\ 0.3 & 0.8 \end{bmatrix} \end{array}, \qquad F(b) = \begin{array}{c} \\ s_1 \\ s_2 \end{array} \begin{array}{c} \begin{array}{cc} s_1 & s_2 \end{array} \\ \begin{bmatrix} 0.2 & 0.7 \\ 0.9 & 0.6 \end{bmatrix} \end{array}.$$

Define 1-FG $= (V_N, V_T, P, \sigma, J, \{f_0, f_1\})$ as follows. $V_N = \{\langle s_1\rangle, \langle s_2\rangle\}$, $V_T = \{a, b\}$, $\sigma = \langle s_1\rangle$, and the 0-fold fuzzy function f_0 is

$$f_0(1) = 0.5, \quad f_0(2) = 0.4, \quad f_0(3) = 0.2, \quad f_0(4) = 0.7,$$

$$f_0(9) = 0.5, \quad f_0(10) = 0.4, \quad f_0(11) = 0.2, \quad f_0(12) = 0.7.$$

The rules with fuzzy vector are

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
(1) $\langle s_1\rangle \to a\langle s_1\rangle$	0.5	0.4	0.2	0.7					0.5	0.4	0.2	0.7				
(2) $\langle s_1\rangle \to a\langle s_2\rangle$					0.3	0.8	0.9	0.6					0.3	0.8	0.9	0.6
(3) $\langle s_1\rangle \to b\langle s_1\rangle$	0.5	0.4	0.2	0.7					0.5	0.4	0.2	0.7				
(4) $\langle s_1\rangle \to b\langle s_2\rangle$					0.3	0.8	0.9	0.6					0.3	0.8	0.9	0.6
(5) $\langle s_2\rangle \to a\langle s_1\rangle$	0.5	0.4	0.2	0.7					0.5	0.4	0.2	0.7				
(6) $\langle s_2\rangle \to a\langle s_2\rangle$					0.3	0.8	0.9	0.6					0.3	0.8	0.9	0.6
(7) $\langle s_2\rangle \to b\langle s_1\rangle$	0.5	0.4	0.2	0.7					0.5	0.4	0.2	0.7				
(8) $\langle s_2\rangle \to b\langle s_2\rangle$					0.3	0.8	0.9	0.6					0.3	0.8	0.9	0.6
(9) $\langle s_1\rangle \to a$								1								
(10) $\langle s_1\rangle \to a$								1								
(11) $\langle s_1\rangle \to b$								1								
(12) $\langle s_1\rangle \to b$								1								
(13) $\langle s_2\rangle \to a$								1								
(14) $\langle s_2\rangle \to a$								1								
(15) $\langle s_2\rangle \to b$								1								
(16) $\langle s_2\rangle \to b$								1								

It is noted that, in the fuzzy vectors from (1) to (8), the values of the blank portions are 0, and the rewriting rule, say, $\langle s_1\rangle \to a$ in (10) is obtained from the rule $\langle s_1\rangle \to a\langle s_2\rangle$ with $s_2 \in G$.

In the sequel we find that n-fold fuzzy grammars n-FG can be transformed into fuzzy automata. Therefore n-fold type 3 fuzzy languages L(n-FG) form a distributive lattice [9] and the language L(n-FG,λ) with the threshold λ is a regular language [8].

REFERENCES

1 M. M. Kherts, *Nauchno-Tekhnicheskaya Informatsya* **2**, No. 1, 29 (1968).
2 T. Huang and K. S. Fu, *Information Sciences* **3**, zol. (1971).
3 A. Salomaa, *Information and Control* **15**, 529 (1969).
4 E. T. Lee and L. A. Zadeh, *Information Sciences* **1**, 421 (1969).
5 J. E. Hopcroft and J. D. Ullman, *Formal Languages and Their Relation to Automata*, Addison-Wesley, Reading, Mass. (1969).
6 L. A. Zadeh, *Information and Control* **8**, 338 (1965).
7 W. G. Wee and K. S. Fu, *IEEE Trans. SSC* **ssc-5**, 215 (1969).
8 E. S. Santos, *Information and Control* **13**, 363 (1968).
9 M. Mizumoto, J. Toyoda, and K. Tanaka, *J. Comp. System Sciences* **3**, 409 (1969).
10 M. Mizumoto, Fuzzy automata and fuzzy grammars, Ph.D. Thesis, Osaka University, February 1971.
11 M. Mizumoto, J. Toyoda, and K. Tanaka, *Information Sciences* **4**, 87 (1972).

Received May 21, 1971