ISSN: 0974-5823 Vol. 6 No. 3 December, 2021

International Journal of Mechanical Engineering

Some Properties of Bounded-Addition Fuzzy Splicing Systems

Mohd Pawiro Santono
Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia,

Mathuri Selvarajoo
Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia,

Wan Heng Fong
Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia

Nor Haniza Sarmin
Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia

***Abstract* - Splicing system is a theoretical model of DNA computing which involves the process of cutting and pasting on DNA molecule with the presence of restriction enzymes and ligase, respectively. When no further control is expected, splicing systems with finite sets of axioms and rules generate only regular languages. Addition of restrictions to the splicing rules can increase the generative power of splicing languages to the recursively enumerable languages. Bounded-addition fuzzy splicing system is introduced in this study. Some fundamental properties of language formed by bounded-addition fuzzy splicing systems were established. The addition of fuzzy as a restriction on splicing operations was studied, and it was discovered that fuzzy splicing systems can increase the generative power of splicing systems with finite components that rely on bounded-addition operations and threshold language cut- points.**

Index Terms – Formal Language Theory, Fuzzy Splicing System, Restriction, Bounded-Addition.

1. INTRODUCTION

Every living organism has a unique deoxyribonucleic acid (DNA). The structure of DNA was firstly introducedin 1953 by Watson and Crick [1] as a double-helical form. DNA molecules are constructed from monomers called nucleotides. Nucleotides have a very simple structure consisting of three components: sugar, phosphate, and base [2]. These structures of DNA were different from each other by the sequence of their bases namely*Adenine, Guanine, Cytosine*, and *Thymine* abbreviated as A, G, C and T respectively. These bases were tied togetherby hydrogen bonds using base-complementary rules, where A pairs with T, G pairs with C and vice versa. These rules ofpairing can simply be written as a, g, c and t [1].

Splicing systems were first proposed by Head in 1987 [3] as a mathematical model of the recombinant behavior of double-stranded DNA (dsDNA) and the enzymes that cut and paste dsDNA. Restriction enzymes, which were found naturally in bacteria, can cut DNA fragments at certain sequences called restriction sites; ligases, on the other hand, can re-join DNA fragments with complementary ends [2]. This model consists of a finite alphabet *V*, a finite set of initial strigs over alphabet *A*, and a finite set of rules *R* that act upon the strings by iterated cutting and pasting, and generates new strings [1]. Splicing language is a language generated by the splicing system. It has been proven that all splicing languages with finite sets of axioms and rules are regular. However, not every regular language is a splicing language. Thus, to increase the generative power of the languages generated by splicing systems, several restrictions are imposed on the splicing systems [4].

Since splicing systems with finite sets of axioms and rules generate only regular languages (see [5]), numerous restrictions on the use of rules have been studied, including probability, group, and weights [4], [6], [7]. All of these restrictions have one thing in common: they can increase the generative power of the language the generated up to recursively enumerable languages. This is significantin terms of DNA computing: restricted splicing systems can be thought of as theoretical models for universally programmable DNA-based computers.

Various problems in computer science and related fields prompted researchers to seek appropriate models for solving these issues. Fuzzy models, for example, have been widely used to develop accurate tools for natural and programming language processing. Fuzzy grammar eliminates ambiguity and leads to more effective language processing parsingand tagging algorithms. The study of fuzzy grammars and fuzzy automata can be found in [8]–[10]. The fact that fuzzy concepts in formal language and automata theories can be applied in DNA computing theory is interesting. The concept of fuzzy splicing systems with multiplication operations was first presented in [11]. It has been demonstrated that fuzzy splicing systems can increase generative power up to context-sensitive languages when applying multiplicationoperations [11].

The purpose of this study is to analyze fuzzy splicing systems with bounded-addition operations, whose grammar and automata counterparts have been extensively studied in recent years. The concept of bounded-addition fuzzy splicing systems is introduced as follows: for each axiom, the truth values are associated with the closed interval [0, 1], and the true value of a string z is obtained by applying a bounded-addition fuzzy operation to the truth values of strings x and y [6], [12]. A threshold language is defined as a subset of the language generated based on some cut-points in [0, 1].

The following is a breakdown of the paper's structure. Section 2 includes several important definitions and notations from formal language, splicing system and fuzzy set ideas. Section 3 proposes the concept of bounded-addition fuzzy splicing systems and threshold languages, provides two examples of bounded-addition fuzzy splicing systems, andestablishes some basic results on the generative power of bounded-addition fuzzy splicing systems. It has been proven that a finite bounded-addition fuzzy splicing system generates not only regular but also context-free and context- sensitive languages. Section 4 concluded the research with a discussion on the overall findings.

2. PRELIMINARIES

In this section, some prerequisites were covered by outlining the basic concepts and notations of the formal language and the splicing systems theories that will be used later. More details can be referred to [13]–[15].

The following general notations are used throughout the paper. The term $\in$ denotes an element's membership in a set, whereas $\notin$ denotes the absence of set membership. The strictness of the inclusion is specified by $\subset$, while $\subseteq$ standsfor (proper) inclusion. The symbol $\varnothing$ represents an emptyset, while $|X|$ represents the cardinality of a set X.

The families of recursively enumerable, context-sensitive, context-free, linear, regular and finite languages were denoted by **RE, CS, CF, LIN, REG** and **FIN** respectively. For these language families, the next strict inclusions, namedChomsky hierarchy (see [14]), holds:

$$\mathbf{FIN} \subset \mathbf{REG} \subset \mathbf{LIN} \subset \mathbf{CF} \subset \mathbf{CS} \subset \mathbf{RE}.$$

Theorem 2.1 [16]: The relations in the following table hold, where at the intersection of the row marked with F_1 with the column marked with F_2 there appear either the family $\mathrm{EH}(F_1, F_2)$ or two families F_3, F_4 such that $F_3 \subset \mathrm{EH}(F_1, F_2) \subseteq F_4$.

Table 1 The family of languages generated by splicing systems

$F_1 \backslash F_2$	**FIN**	**REG**	**LIN**	**CF**	**CS**	**RE**
FIN	**REG**	**RE**	**RE**	**RE**	**RE**	**RE**
REG	**REG**	**RE**	**RE**	**RE**	**RE**	**RE**
LIN	**LIN, CF**	**RE**	**RE**	**RE**	**RE**	**RE**
CF	**CF**	**RE**	**RE**	**RE**	**RE**	**RE**
CS	**RE**	**RE**	**RE**	**RE**	**RE**	**RE**
RE	**RE**	**RE**	**RE**	**RE**	**RE**	**RE**

When proposing fuzzy sets, Zadeh's (1965) concerns were explicitly centered on their potential contribution in the domains of pattern classification, processing, and communication of information, abstraction and summarization [8]. In the addition of fuzzy as a restriction, the fuzzy membership values from close intervals [0, 1] will be assigned to the axioms of the splicing system. Then, the truth values of every generated string will be calculated using fuzzy operation over their fuzzy membership values and to determine the classes of languages generated by the fuzzy splicing system [11].

3. MAIN RESULTS

In this section, the concept of bounded-addition fuzzy splicing system was introduced by first assigning truth values(i.e., fuzzy membership values) to the axioms of splicing systems from the closed interval [0, 1]. Then, using a fuzzy bounded-addition operation over the truth values of strings $x$and y, the truth value of each created string x is calculated.

Definition 3.1: A bounded-addition fuzzy splicing system is a 6-tuple $\gamma^{\oplus}=(V,T,A^{\oplus},R,\mu,\oplus)$ where *V, T, R* are defined as for a usual extended splicing systems, $\mu : V^{*}\times[0,1]$ is a fuzzy membership function, $A^{\oplus}$ is a subset of $V^{*}\times[0,1]$ such that

$$\sum_{i=1}^{n}\mu(x_i)\le 1$$

and $\oplus$ is a bounded-addition fuzzy operation on [0, 1] defined by

$$\mu_{A+B}=\mu_A+\mu_B-\mu_A\mu_B.$$

A fuzzy bounded-addition operation is defined next.

Definition 3.2: For strings with fuzzy $(x,\mu(x))$, $(y,\mu(y)),\,(z,\mu(z))\in V^{*}\times[0,1]$ and $r\in R$ the fuzzy bounded-addition operation is defined as

$$[(x,\mu(x)),(y,\mu(y))]\ \text{a}\ \ _{r}\ z(z,\mu(z))$$

if and only if $(x,y)\ \text{a}\ \ _{r}\ z$ and $\mu(z)=\mu(x)\oplus\mu(y)$ is defined by

$$\mu_{x+y}=\mu_x+\mu_y-\mu_x\mu_y \text{ where } \mu_x\mu_y\in\mu(x_i).$$

Moreover, a fuzzy bounded-addition operation on $A^{*}\times[0,1]$ as well as an iterated fuzzy splicing operation was defined as in the following.

Definition 3.3: Let $\gamma^{\oplus}=(V,T,A^{\oplus},R,\mu,\oplus)$ be a bounded-addition fuzzy splicing system. Then

$$\sigma(A^{\oplus})=\{(z,\mu(z):(x,y)\ \text{a}\ \ _{r}\ z\wedge\mu(z)=\mu(x)\oplus\mu(y)$$

for some $(x,\mu(x)),(y,\mu(y))\in A^{\oplus}$ and $r\in R$,and the iterative splicing operation is defined as

$$\sigma^{*}(A^{\oplus})=\bigcup_{i\ge 1}\sigma^{i}(A^{\oplus})$$

where

$$\sigma^{0}(A^{\oplus})=A^{\oplus},$$

$$\sigma^{i}(A^{\oplus})=\sigma^{i-1}(A^{\oplus})\cup\sigma(\sigma^{i-1}(A^{\oplus})) \text{ for } i\ge 0.$$

Definition 3.4: The language generated by iterative bounded-addition fuzzy splicing system $\gamma^{\oplus}=(V,T,A^{\oplus},R,\mu,\oplus)$ is defined as

$$L_f(\gamma^{\oplus})=\{x\in T\mid(x,\mu(x))\in\sigma^{*}(A^{\oplus})\}.$$

Remark 3.1: Different splicing operations may result in the same string with different fuzzy memberships. Weak and strict threshold languages were considered in order to avoid this "ambiguity". In strict language, all of a string's fuzzy membership values must fulfil the threshold modes that will be defined. While in the weak language, satisfying just one string membership value was sufficient.

Next, the properties of bounded-addition fuzzy splicing system with threshold points are defined.

Definition 3.5: Let $L_f(\gamma^{\oplus})$ be the language generated by a bounded-addition fuzzy extended splicing system $\gamma^{\oplus}=(V,T,A^{\oplus},R,\mu,\oplus).$ The thresholds (cut-points) are considered as subsegments and discrete subsets of [0, 1] as well as real numbers in [0, 1]. The strict and weak thresholdlanguages with respect to thresholds $\Phi\subseteq[0,1]$ and $\varphi\in[0,1]$ are defined as follows:

$$L_f(\lambda^{\oplus},\Diamond,\varphi)=\{z\in T^{*}\mid(x,\mu(x))\in\sigma^{*}(A^{\oplus}) \text{ and for all } \mu(x),\mu(x)\ \Diamond\ \varphi\}$$
$$L_f(\lambda^{\oplus},*,\Phi)=\{z\in T^{*}\mid(x,\mu(x))\in\sigma^{*}(A^{\oplus}) \text{ and for all } \mu(x),\mu(x) * \Phi\}$$
$$L_f(\lambda^{\oplus},\Diamond,\varphi)=\{z\in T^{*}\mid(x,\mu(x))\in\sigma^{*}(A^{\oplus}) \text{ and for some } \mu(x),\mu(x)\ \Diamond\ \varphi\}$$
$$L_f(\lambda^{\oplus},*,\Phi)=\{z\in T^{*}\mid(x,\mu(x))\in\sigma^{*}(A^{\oplus}) \text{ and for some } \mu(x),\mu(x) * \Phi\}$$

where $\Diamond\in\{=,\ne,\ge,>,\le,<\}$ and $*\in\{\in,\notin\}$ are called threshold modes.

The family of threshold languages generated by bounded-addition fuzzy extended H systems of type (F_1, F_2) were denoted by $f^{\oplus}\mathrm{EH}(F_1,F_2)$, where $F_1,F_2\in\{\mathbf{FIN},\mathbf{REG},\mathbf{CF},\mathbf{LIN},\mathbf{CS},\mathbf{RE}\}.$

Lemma 3.1: For all families $F_1,F_2\in\{\mathbf{FIN},\mathbf{REG},\mathbf{CF},\mathbf{LIN},\mathbf{CS},\mathbf{RE}\}.$

$$\mathrm{EH}(F_1,F_2)\subseteq f^{\oplus}\mathrm{EH}(F_1,F_2)$$

Proof: Let $\gamma=(V,T,A,R)$ be extended splicing system generating the language $L(\gamma)\in \mathrm{EH}(\mathbf{FIN},\ \mathrm{F})$ where $F_1,F_2\in\{\mathbf{FIN, REG, CF, CS, RE}\}$. Let A = {x$_1$, x$_2$,…, x$_n$}, n ≥ 1. A bounded-addition fuzzy splicing system is defined by $\gamma^{\oplus}=(V,T,A^{\oplus},R,\mu,\oplus)$ where the set of axioms is defined by

$$A^{\oplus}=\{(x_i,\mu(x_i)):x_i\in A^{\oplus},1\le n\}$$

where μ(x$_i$)=1/n for all 1 ≤ n, then

$$\sum_{i=1}^{n}\mu(x_i)\le 1$$

and $\oplus$ is a bounded-addition fuzzy operation on [0, 1] defined by

$$\mu_{A+B}=\mu_A+\mu_B-\mu_A\mu_B \text{ where } \mu_A\mu_B\in\mu(x_i).$$

The threshold language generated by $\gamma^{\oplus}$ is defined as $L_f(\gamma^{\oplus}>0),$ then it is clear that $L(\gamma)=L_f(\gamma^{\oplus}>0).$ Hence, $\mathrm{EH}(F_1,F_2)\subseteq f^{\oplus}\mathrm{EH}(F_1,F_2)$ for all families $F_1,F_2\in\{\mathbf{FIN, REG, CF, LIN, CS, RE}\}.$ ■

The following examples demonstrate that with this restriction, the generative power of bounded-addition fuzzy splicing systems can be increased up to the context-sensitive languages.

Example 3.1: Consider the bounded-addition fuzzy splicing system

$$\gamma_1^{\oplus}=(\{a,b,c,d\},\{a,b,c\},\{(cad,\frac{1}{3}),(dbc,\frac{1}{2})\},R,\mu,\oplus)$$

where

$$R=\{r_1,r_2,r_3\} \text{ and } r_1=a\#d\$c\#a,$$

$$r_2=b\#c\$d\#b \text{ and } r_3=a\#d\$d\#b$$

When the first rule r_1 is applied in a string *cad*, the stringobtained is

$$[(cad,\frac{1}{3}),(cad,\frac{1}{3})]\ \text{a}_{r_1}\ (caad,\frac{5}{9}).$$

By iterative splicing operation between the same string using the rule r_1, the string

$$(ca^nd,\frac{2^n}{3^n})$$

is obtained.

By applying the rule r_2 to the string *dbc*, the string obtained is

$$[(dbc,\frac{1}{2}),(dbc,\frac{1}{2})]\ \text{a}_{r_2}\ (dbbc,\frac{3}{4}).$$

By iterative splicing operation between the same string using the rule r_2, the string

$$(db^mc,1-\frac{1}{2^m})$$

is obtained.

From the strings ca^nd, $n\ge1$, and db^mc, $m\ge1$, by using the rule r_3, the iterative splicing is

$$(ca^nd,\frac{2^n}{3^n}),(db^mc,1-\frac{1}{2^m})\ \text{a}_{r_3}\ (ca^nb^mc,1-\frac{2^n}{3^n2^m}).$$

Thus,

$$L_f(\gamma_1^{\oplus})=\{(ca^nb^mc,1-\frac{2^n}{3^n2^m})\mid k\ge1,m\ge1\}.$$

When bounded-addition fuzzy splicing systems with different thresholds and modes are considered, the threshold languages generated are

1. $L_f(\gamma_1^{\oplus}=1-\frac{2^n}{3^n2^m})=\{ca^nb^mc\mid m,n\ge1\}\in\mathbf{REG,}$
2. $L_f(\gamma_1^{\oplus}<\frac{2}{3})=\varnothing\in\mathbf{FIN,}$
3. $L_f(\gamma_1^{\oplus}=\frac{2}{3})=\{cabc\}\in\mathbf{FIN,}$
4. $L_f(\gamma_1^{\oplus}=1-(\frac{1}{3})^n)=\{ca^nb^nc\mid n\ge1\}\in\mathbf{CF\text{-}REG.}$

As it can be seen, the last threshold language generated by the bounded-addition fuzzy splicing system was not regular. However, if the bounded-addition was replaced with *min* or *max*, the threshold languages were not more than regular. In this case, for the generated strings ca^nb^mc,

$$\mu(ca^nb^mc)=\begin{cases}\frac{1}{3},m,n>0,\\ \frac{1}{2},m,n=0.\end{cases}$$

Therefore,

$$L_f(\gamma_1^{\oplus}=1-(\frac{1}{3})^n;n\ge1)=\varnothing\in\mathbf{FIN,}$$

$$L_f(\gamma_1^{\oplus}\ge\frac{1}{3})=\{ca^nb^mc\mid n,m\ge1\}\in\mathbf{REG.}$$

Example 3.2: Consider the bounded-addition fuzzy splicing system

$$\gamma_2^{\oplus} = (\{a,b,c,w,x,y,z\},\{a,b,c,w,x\},\{(xay,\frac{1}{3}),$$

$$(ybz,\frac{1}{5}),(zcw,\frac{1}{7})\},\{r_1,r_2,r_3,r_4,r_5\},\mu,\oplus)$$

where

$$r_1 = xa\#y\$x\#a, r_2 = yb\#z\$y\#b, r_3 = zc\#w\$z\#c$$

$$r_4 = a\#y\$y\#b \text{ and } r_5 = b\#z\$z\#c.$$

When the first rule r_1 is applied in a string *xay*, by iterative splicing operation between the same string using the same rule, the string

$$(xa^k y, 1-\frac{2^k}{3^k}), k \geq 1$$

is obtained.

When the second rule r_2 is applied in a string *ybz*, by iterative splicing operation between the same string using the same rule, the string

$$(yb^m z, 1-\frac{4^m}{5^m}), m \geq 1$$

is obtained.

When the second rule r_3 is applied in a string *zcw,* by iterative splicing operation between the same string using the same rule, the string

$$(zc^n w, 1-\frac{6^n}{7^n}), n \geq 1$$

is obtained.

The non-terminals *y* and *z* from these strings are eliminated by rule r_4 and r_5.

$$(xa^k y, 1-\frac{2^k}{3^k}),(yb^m z, 1-\frac{4^m}{5^m}) \text{ a }_{r_4}$$

$$(xa^k b^m z, 1-\frac{2^k 4^m}{3^k 5^m})$$

and

$$(xa^k b^m z, 1-\frac{2^k 4^m}{3^k 5^m}),(zc^n w, 1-\frac{6^n}{7^n}) \text{ a }_{r_5}$$

$$(xa^k b^m c^n w, 1-\frac{2^k 4^m 6^n}{3^k 5^m 7^n}).$$

Then, the language generated by the bounded-addition fuzzy splicing system $\gamma_2^{\oplus}$,

$$L_f(\gamma_2^{\oplus}) = \{(xa^k b^m c^n w, 1-\frac{2^k 4^m 6^n}{3^k 5^m 7^n}) \mid k,m,n \geq 1\}.$$

Further, the following threshold languages were considered:

1. $L_f(\gamma_2^{\oplus} = 1-\frac{2^k 4^m 6^n}{3^k 5^m 7^n}) = \{xa^k b^m c^n w \mid k,m,n \geq 1\}$ $\in$ **REG,**
2. $L_f(\gamma_2^{\oplus} < \frac{19}{25}) = \varnothing \in$ **FIN,**
3. $L_f(\gamma_2^{\oplus} = \frac{19}{25}) = \{xabcw\} \in$ **FIN,**
4. $L_f(\gamma_2^{\oplus} = 1-(\frac{16}{35})^n) = \{xa^n b^n c^n w \mid n \geq 1\} \in$ **CS - CF.**

The examples above illustrate that the use of thresholds with bounded-addition fuzzy splicing system increased the generative power of splicing systems with finite components. Two simple but interesting facts of bounded-addition fuzzy splicing systems are mentioned as Corollary 3.1 andCorollary 3.2 as stated in the following.

Corollary 3.1: If the fuzzy membership of each axiom $\mu(x) \in A^{\oplus}$ in a bounded-addition fuzzy splicing system $\gamma^{\oplus} = (V,T,A^{\oplus},R,\mu,\oplus)$ is nonzero, then the threshold language $L_f(\gamma^{\oplus} < \alpha)$ with $\alpha \in [0,1]$ is an empty set, i.e., $L_f(\gamma^{\oplus} < \alpha) = \varnothing.$

Corollary 3.2: If the fuzzy membership of each axiom $\mu(x) \in A^{\oplus}$ in a bounded-addition fuzzy splicing system $\gamma^{\oplus} = (V,T,A^{\oplus},R,\mu,\oplus)$ is not greater than 1, then every threshold language $L_f(\gamma^{\oplus} = \alpha)$ with $\alpha \in [0,1]$ is finite.

From Theorem 2.1, Lemma 3.1 and Examples 3.1 and 3.2, the following two theorems were obtained.

Theorem 3.1: Let $\gamma^{\oplus}=(V,T,A^{\oplus},R,\mu,\oplus)$ be a bounded-addition fuzzy extended splicing system, where $0<\mu(x)<1$ for all $x\in A^{\oplus}$ and $\alpha\in[0,1]$. Then,

1. $L_f(\gamma^{\oplus}>\alpha)$ is a finite language.
2. $L_f(\gamma^{\oplus}\le\alpha)$ is a regular language.
3. $L_f(\gamma^{\oplus}\in I)$ is a regular language where *I* is a subsegment of [0, 1].

Proof:

Case 1: Let $\gamma^{\oplus}=(V,T,A^{\oplus},R,\mu,\oplus)$ be a bounded-addition fuzzy extended splicing system where

$$A^{\oplus}=\{(x_1,\mu(x_1)),(x_2,\mu(x_2)),...,(x_n,\mu(x_n))\}$$

and $\mu(x_i)=\mu_i,\ 1\le i\le n.$ Since $0<\mu(x_i)<1,$

$$\sum_{j=1}^{k+1}\mu(x_{ij})>\sum_{j=1}^{k}\mu(x_{ij}),\mu_{ij}\in\{\mu_1,...,\mu_n\}.$$

Then, there exist a positive integer $m = k + 1\in N$ such that

$$\sum_{j=1}^{m}\mu(x_{ij})<\alpha,\mu_{ij}\in\{\mu_1,...,\mu_n\}$$

where $1\le j\le m$.

For any string $x\in\sigma_f^{i}(A^{\oplus}),\ i\ge m$ that was obtained from some strings of $\sigma_f^{i-1}(A^{\oplus})$ using more than or equal to *m* splicing operations, $\mu(x)<\alpha$ is produced. Thus, $L_f(\gamma^{\oplus}>\alpha)$ contains a finite number of strings.

Case 2: Let $L_f(\gamma^{\oplus})=L_f(\gamma^{\oplus}>\alpha)\cup L_f(\gamma^{\oplus}\le\alpha).$ Since $L_f(\gamma^{\oplus})$ is regular and $L_f(\gamma^{\oplus}>\alpha)$ is finite, then $L_f(\gamma^{\oplus}\le\alpha)$ is regular.

Case 3: If $I=(\alpha_1,\alpha_2),$ then $L_f(\gamma^{\oplus}\in I)=$ $L_f(\gamma^{\oplus}>\alpha)\cap L_f(\gamma^{\oplus}<\alpha).$ Hence, according to ***Case 1*** and ***Case 2***, $L_f(\gamma^{\oplus}\in I)$ is regular. ■

Theorem 3.2: Let $\gamma^{\oplus}=(V,T,A^{\oplus},R,\mu,\oplus)$ be a bounded-addition fuzzy extended splicing system and $L_f(\gamma^{\oplus}*\alpha)$ be a threshold language where $\oplus\in\{min,max\},\ *\in\{>,<,=\}$ and $\alpha\in[0,1].$ Then,

1. $L_f(\gamma^{\oplus}*\alpha)$ is a regular language,
2. If α is *max*, then $L_f(\gamma^{\oplus}>\alpha)=\varnothing$ and $L_f(\gamma^{\oplus}\le\alpha)$ $=L_f(\gamma^{\oplus}),$
3. If α is *min*, then $L_f(\gamma^{\oplus}\le\alpha)=\varnothing$ and $L_f(\gamma^{\oplus}>\alpha)$ $=L_f(\gamma^{\oplus}),$
4. If *I* is a subsegment of [0, 1], then $L_f(\gamma^{\oplus}\in I)$ is a regular languages.

Proof:

Case 1: Let $\gamma^{\oplus}=(V,T,A^{\oplus},R,\mu,\oplus)$ be a bounded-addition fuzzy extended splicing system with

$$A^{\oplus}=\{(x_1,\mu(x_1)),(x_2,\mu(x_2)),...,(x_n,\mu(x_n))\}.$$

Consider *max* as a bounded-addition fuzzy operation and > as a threshold mode. Then, the set of $\sigma_f^{*}(A^{\oplus})$ can be represented as $\sigma_f^{*}(A^{\oplus})=\sigma_{f1}^{*}(A^{\oplus})\cup\sigma_{f2}^{*}(A^{\oplus})$ where

$$\sigma_{f1}^{*}(A^{\oplus})=\{(x,\mu(x))\in\sigma_{f1}^{*}(A^{\oplus}):\mu(x)>\alpha\},$$

and

$$\sigma_{f2}^{*}(A^{\oplus})=\{(x,\mu(x))\in\sigma_{f1}^{*}(A^{\oplus}):\mu(x)\le\alpha\},$$

Let $\sigma_{f1}^{*}(A^{\oplus})=A_i^{\oplus},i=1,2.$ Then, $A^{\oplus}=A_1^{\oplus}\cup A_2^{\oplus}$ where

$$A_1^{\oplus}=\{x\in A^{\oplus}:\mu(x)>\alpha\},\text{ and}$$

$$A_2^{\oplus}=\{x\in A^{\oplus}:\mu(x)\le\alpha\}.$$

The splicing system $\gamma=(V,T,A_2,R)$ is constructed where $L(\gamma)=\alpha^{*}(A_2)\cap T^{*}$ is regular. Moreover, it is shown that $\sigma_{f2}^{*}(A^{\oplus})=\sigma_f^{*}(A_2^{\oplus})$. First, $\sigma_f^{*}(A_2^{\oplus})\subseteq\sigma_{f2}^{*}(A^{\oplus})$ since $A_2^{\oplus}\subseteq A^{\oplus}$. On the other hand, $\sigma_{f2}^{*}(A^{\oplus})\subseteq\sigma_f^{*}(A_2^{\oplus}).$ Let $x\notin\sigma_f^{*}(A_2^{\oplus}).$ Then, there is an axiom $(x,\mu(x))\in A_1^{\oplus}$ such that

$$((x_1,\mu(x_1)),(x_2,\mu(x_2)))\mapsto(z_1,\mu(z_1)),$$
$$((z_1,\mu(z_1)),(z_2,\mu(z_2)))\mapsto(z_3,\mu(z_3)),$$
$$\vdots$$
$$((z_k,\mu(z_k)),(z_{k+1},\mu(z_{k+1})))\mapsto(x,\mu(x)),$$

where $(x_2,\mu(x_2))\in A^{\oplus}$ and $(z_1,\mu(z_1))\in\sigma_f^{*}(A^{\oplus}).$ Then,

$$max\{\mu(x_1),\mu(x_2)\}=\mu(z_1)>\alpha,$$
$$\vdots$$
$$max\{\mu(z_{k+1}),\mu(z_{k+1})\}=\mu(z_1)>\alpha.$$

Consequently, $(x,\mu(x))\notin\sigma_{f2}^{*}(A^{\oplus})$. Thus, $\sigma_{f2}^{*}(A^{\oplus})$ $=\sigma_{f}^{*}(A_{2}^{\oplus})$. It follows that the language $L_f(\gamma^{\oplus}\le\alpha)$ $=\sigma_{f2}^{*}(A^{\oplus})\cap T^{*}$ is regular. Hence, $\sigma_{f1}^{*}(A^{\oplus})=$ $\sigma_{f}^{*}(A^{\oplus})\setminus\sigma_{f2}^{*}(A^{\oplus})$ and the language $L_f(\gamma^{\oplus}>\alpha)=$ $L_f(\gamma^{\oplus})\setminus L_f(\gamma^{\oplus}\le\alpha)$ is also regular. Similarly, if the bounded-addition fuzzy operation is *min*, it can be proven that $L_f(\gamma^{\oplus}>\alpha)$ and $L_f(\gamma^{\oplus}\le\alpha)$ are regular.

Case 2: Let $\alpha>max\{\mu_1,\mu_2,...,\mu_n\}$. Based on ***Case 1***, *max* as bounded-addition fuzzy operation was considered. Then, it produces an axiom such that

$$max\{\mu(x_1),\mu(x_2)\}=\mu(z_1)<\alpha,$$

M

$$max\{\mu(z_{k+1}),\mu(z_{k+1})\}=\mu(z_1)<\alpha.$$

Thus, the language $L_f(\gamma^{\oplus}\le\alpha)$ is regular and $L_f(\gamma^{\oplus}>\alpha)$ is an empty set produced. The language $L_f(\gamma^{\oplus})=L_f(\gamma^{\oplus}>\alpha)\cup L_f(\gamma^{\oplus}\le\alpha)$. Since, $L_f(\gamma^{\oplus}>\alpha)$ is empty and $L_f(\gamma^{\oplus}\le\alpha)$ is regular, then $L_f(\gamma^{\oplus})$ is regular. Hence, $L_f(\gamma^{\oplus})=L_f(\gamma^{\oplus}\le\alpha)$.

Case 3: Let $\alpha>min\{\mu_1,\mu_2,...,\mu_n\}$. Based on ***Case 1***, *min* as bounded-addition fuzzy operation was considered. Then, it produces an axiom such that

$$min\{\mu(x_1),\mu(x_2)\}=\mu(z_1)>\alpha,$$

M

$$min\{\mu(z_{k+1}),\mu(z_{k+1})\}=\mu(z_1)>\alpha.$$

Thus, the language $L_f(\gamma^{\oplus}>\alpha)$ is regular and $L_f(\gamma^{\oplus}\le\alpha)$ is an empty set produced. The language $L_f(\gamma^{\oplus})=L_f(\gamma^{\oplus}>\alpha)\cup L_f(\gamma^{\oplus}\le\alpha)$. Since, $L_f(\gamma^{\oplus}\le\alpha)$ is empty and $L_f(\gamma^{\oplus}>\alpha)$ is regular, then $L_f(\gamma^{\oplus})$ is regular. Hence, $L_f(\gamma^{\oplus})=L_f(\gamma^{\oplus}>\alpha)$.

Case 4: Let $L_f(\gamma^{\oplus}\in I)=L_f(\gamma^{\oplus}>\alpha_1)\cap L_f(\gamma^{\oplus}$ $<\alpha_2)$ where $I=(\alpha_1,\alpha_2)$. From ***Case 1***, $L_f(\gamma^{\oplus}>\alpha_1)$ and $L_f(\gamma^{\oplus}<\alpha_2)$ are regular. Therefore, their intersections are also regular. ■

From the theorems above, the following corollary was obtained.

Corollary 3.3: Every fuzzy splicing system with the bounded-addition operation, *max* or *min*, and the cut-points of any number in [0, 1] or any subinterval of [0, 1] generate a regular language.

4. CONCLUSION

The concept of bounded-addition fuzzy splicing systems has been proposed and its preliminary properties were established in this study. The truth values were associated from closed interval [0, 1] with each axiom, and the truth value of astring *z* generated from strings *x* and *y* are calculated by applying fuzzy bounded-addition operation over the truth value. It shows that an extension of splicing systems by introducing bounded-addition fuzzy splicing systems increases the generative power of splicing systems with finite components up tosome context-sensitive languages. Besides, some threshold languages with the selection of appropriate cut-points also can generate non-regular languages.

5. ACKNOWLEDGEMENTS

The first and second authors would like to thank the Ministryof Higher Education and Universiti Teknologi MARA Malaysia for the financial funding through research grant FRGS-RACER (600-IRMI/FRGS-RACER 5/3 (050/2019)). The third and fourth authors would also like to thank the Ministry of Higher Education and Research Management Center (RMC) UTM for the UTMShine Grant Vote No. Q.J130000.2454.09G89.

REFERENCES

[1] Y. Yusof, N. H. Sarmin, T. E. Goode, M. Mahmud, and F. W. Heng, "An extension of DNA splicing system," *Proc. - 2011 6th Int. Conf. Bio-Inspired Comput. Theor. Appl. BIC-TA 2011*, pp. 246–248, 2011, doi: 10.1109/BIC-TA.2011.67.

[2] M. Amos, G. Pun, G. Rozenberg, and A. Salomaa, "Topics in the theory of DNA computing," *Theor. Comput. Sci.*, vol. 287, no. 1, pp. 3–38, 2002, doi: 10.1016/S0304-3975(02)00134-2.

[3] G. Păun, "On the splicing operation," *Discret. Appl. Math.*, vol. 70, no. 1, pp. 57–79, 1996, doi: 10.1016/0166-218X(96)00101-1.

[4] N. Z. A. Hamzah, N. A. Mohd Sebry, W. H. Fong, N. H. Sarmin, and S. Turaev, "Splicing Systems over Permutation Groups of Length Two," *Malaysian J. Fundam. Appl. Sci.*, vol. 8, no. 2, pp. 83–88, 2014, doi: 10.11113/mjfas.v8n2.127.

[5] L. Kari and S. Kopecki, "Deciding whether a regular language is generated by a splicing system," *J. Comput. Syst. Sci.*, vol. 84, pp. 263–287, 2017, doi: 10.1016/j.jcss.2016.10.001.

[6] N. T. Nguyen, B. Trawiński, R. Katarzyniak, and G. S. Jo, "Probabilistic Splicing Systems," *Stud. Comput. Intell.*, vol. 457, no. January, 2013, doi: 10.1007/978-3-642-34300-1.

[7] S. Turaev, Y. S. Gan, M. Othman, N. H. Sarmin, and W. H. Fong, "Weighted splicing systems," in *Communications in Computer and Information Science*, 2012, vol. 316 CCIS, pp. 416–424, doi: 10.1007/978-3-642-34289-9_46.

[8] K. S. Ray and M. Mondal, "Splicing operation and fuzzy molecular automaton," *Int. J. Intell. Comput. Cybern.*, vol. 5, no. 2, pp. 206–225, 2012, doi: 10.1108/17563781211231543.

[9] C. Moraga, “An approach to fuzzy context free languages,” no. November, pp. 17–19, 2008.

[10] M. Inui, W. Shoaff, L. Fausett, and M. Schneider, “The recognition of imperfect strings generated by fuzzy context sensitive grammars,” *Fuzzy Sets Syst.*, vol. 62, no. 1, pp. 21–29, 1994, doi: 10.1016/0165-0114(94)90069-8.

[11] F. Karimi, S. Turaev, N. H. Sarmin, and W. H. Fong, “Fuzzy splicing systems,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8733, pp. 20–29, 2014, doi: 10.1007/978-3-319-11289-3_3.

[12] M. Mizumoto and K. Tanaka, “Fuzzy sets and their operations,” *Inf. Control*, vol. 48, no. 1, pp. 30–48, 1981, doi: 10.1016/S0019-9958(81)90578-7.

[13] J. E. Hopcroft, R. Motwani, Rotwani, and J. D. Ullman, “Introduction to Automata Theory, Languages and Computability,” 2000.

[14] G. Rozenberg and A. Salomaa, *Handbook of Formal Languages*, no. January. 1997.

[15] N. H. Sarmin, Y. Yusof, and F. Wan Heng, “Some characterizations in splicing systems,” *AIP Conf. Proc.*vol. 1309, no. December, pp. 411–418, 2010, doi: 10.1063/1.3525142.

[16] G. Păun, G. Rozenberg, and A. Salomaa, “DNA computing: New computing paradigms,” *Comput. Math. with Appl.*, vol. 37, no. 3, p. 134, 1999, doi: 10.1016/s0898-1221(99)90411-x.