J. Appl. Math. & Informatics Vol. 42(2024), No. 3, pp. 483 - 497
https://doi.org/10.14317/jami.2024.483

PICTURE PROCESSING ON ISOMETRIC FUZZY REGULAR ARRAY LANGUAGES

A. JOHN KASPAR, D.K. SHEENA CHIRISTY* AND D.G. THOMAS

Abstract. Isometric array grammar is one of the simplest model to generate picture languages, since both sides of its production rule have the same shape. In this paper, we have introduced isometric fuzzy regular array grammars to generate isometric fuzzy regular array languages and discussed its closure properties. Also, the relation between isometric fuzzy regular array grammar and boustrophedon fuzzy finite automata has been discussed. Moreover, we study the relation between two dimensional fuzzy regular grammars with returning fuzzy finite automata and boustrophedon fuzzy finite automata. Further, the hierarchy results of these three classes of languages have been discussed.

AMS Mathematics Subject Classification : 68Q45, 03E72.
Key words and phrases : Boustrophedon fuzzy finite automata, two dimensional fuzzy regular languages, isometric regular array grammars, picture languages.

1. Introduction

A two dimensional word also termed as a picture (matrix, array), is represented by a rectangular array of symbols taken from a finite alphabet. A two dimensional language or a picture language is a collection of pictures. Many researchers investigated the properties and complexity results of two dimensional automata, which operate on two dimensional words. The deep survey of various classes of two dimensional languages and their properties can be found in [4, 5, 7, 15, 17, 18, 20]. Among these classes, the Siromoney matrix model [21] and isotonic (isometric) array grammars introduced in [14] are the simplest models to describe picture languages. The Siromoney matrix model [21] and its extension of array grammars and array automata are very useful models in the generation of picture languages. The advantages of these models are described

Received November 7, 2022. Revised October 26, 2023. Accepted December 22, 2023.
*Corresponding author.

in [18, 19]. On the other hand, isometric array grammars introduced by Rosenfeld [14], have the advantage that each side of the production rules has the same shape. The hierarchy results and properties of isometric array languages have been discussed and studied in [3, 15, 22]. H. Fernau et al.[4] introduced one of the elegant model called boustrophedon finite automata (BFA) for two dimensional languages. The name Boustrophedon is derived from the ancient Greek word which means "as the ox turns". The BFA processes the input-bordered arrays, with the head moving one after the other on every computation step and changing its direction only when the borders of the input array are visited. They have introduced returning finite automata (RFA) and proved its equivalence with BFA. They discussed the relation between regular matrix languages [21] and isometric array languages [14] with BFA. Also, the possible applications of the BFA model is discussed.

Recently, an extension of one dimensional fuzzy regular languages to two dimensional (picture) fuzzy regular languages (2-FRLs) has been introduced in [8], where the basic concepts of fuzzy languages are extended to the picture languages introduced by R. Siromoeny and G. Siromoney [21]. Also, discussed the application of 2-FRLs in the generation geometric and asanapalakai kolam patterns. The concept of fuzzy sets was first found by Zadeh to deal with real-life problems having uncertainties, vagueness and imprecise data [25, 10]. The notion of fuzzy automata was initiated by Wee in [24], which is the generalisation of finite automata [6, 12]. Wee and Fu [23] introduced the mathematical formulation of fuzzy finite automata. Zadeh and Lee initiated fuzzy languages generated by fuzzy grammars to reduce the vagueness that occurs in formal languages [6, 12]. The depth knowledge about various types of fuzzy languages and fuzzy automata can be seen in [1, 2, 11, 13, 16]. In [9], introduced boustrophedon fuzzy finite automata (BFFA), returning fuzzy finite automata (RFFA) and demonstrated how a picture is processed by BFFA and RFFA. The concept of pumping lemma and interchanging lemma for the languages accepted by the BFFA has been studied. Also, discussed the closure properties such as union, intersection, complementation and concatenation of the accepted fuzzy languages of BFFA.

Motivated by the work presented in [4], in this paper we introduce isometric fuzzy regular array languages generated by isometric fuzzy regular grammars and discuss their closure properties. We also study their connection with RFFA as well as with BFFA. Furthermore, discuss the relation of BFFA with two dimensional fuzzy regular languages [8].

This paper is organized as follows: The basic notions of two dimensional languages (regular matrix languages and isometric array languages), two dimensional fuzzy regular languages, boustrophedon fuzzy finite automata and retruning fuzzy finite automata are recalled in Section 2. In Section 3, isometric fuzzy regular array grammars and languages generated by it have been introduced and studied its equivalence with BFFA. The closure properties of isometric fuzzy regular array languages have been discussed. The equivalence of 2-FRL with RFFA and BFFA is discussed in Section 4. In Section 5, we study the hierarchy results of two dimensional fuzzy regular languages, isometric fuzzy regular array languages and the languages accepted by boustrophedon fuzzy finite autoamata.

2. Preliminaries

The basics of regular matrix grammar, regular matrix languages and their properties of corresponding class of picture languages can be found in [17, 21]. Isotonic (isometric) array grammar, isometric array languages and the properties of isometric array languages are recalled from [4, 14, 15]. The notions of two dimensional fuzzy right linear grammar (2-FRLG) and their properties can be found in [8]. Boustrophedon fuzzy finite automata, returning fuzzy finite automata and their results can be recalled from [9].

Definition 2.1. [8]

A *two dimensional fuzzy right - linear grammar* (2-FRLG) is a two-tuple $G = (G_h,\ G_v)$, where G_h and G_v are fuzzy right - linear grammars. Now,

- $G_h = (V_h,\ I_h,\ P_h,\ S_h)$,
  - V_h represents a collection of finite *horizontal non-terminals*,
  - I_h represents a collection of finite *intermediates*,
  - S_h represents a finite fuzzy subset of start symbols from V_h,
  - P_h represents a collection of finite fuzzy horizontal production rules of the form $\alpha \xrightarrow{\rho} x\beta$ or $\alpha \xrightarrow{\rho} x$, where $\rho \in (0, 1]$ is the grade of membership of the production rules, $\alpha, \beta \in V_h,\ x \in I_h^*$ and $V_h \cap I_h = \emptyset$. and
- $G_v = \bigcup\limits_{i=1}^{k} G_{v_i}$ where, $G_{v_i} = (V_{v_i},\ I_v,\ P_{v_i},\ S_i)$, $i = 1,\ 2,\ \ldots,\ k$,
  - V_{v_i} represents a collection of finite *vertical non-terminals* such that $V_{v_i} \cap V_{v_j} = \emptyset$ for $i \neq j$,
  - I_v represents a collection of finite *terminals*,
  - $S_i \in V_{v_i}$ represents the *fuzzy start symbol* and
  - P_{v_i} are represents a collection of finite fuzzy vertical production rules of the form $A \xrightarrow{\sigma} cB$ or $A \xrightarrow{\sigma} c$, $A,\ B \in V_{v_i}$ and $c \in I_v$, where $\sigma \in (0, 1]$ is membership grade of the production rules.

Derivation of the matrix from the given 2-FRLG and the degree of membership of the matrix are given in [8].

Definition 2.2. [8]

A $m \times n$ matrix W is said to be generated by the 2-FRLG G, if,

$\mathcal{S}_G(W) = \max\limits_{S \in S_h}\{\min\{(\mathcal{S}(S) \wedge \mathcal{S}(S \stackrel{*}{\Rightarrow} y)), \{\bigvee\limits_{i=1}^{k}(\mathcal{S}(S_i) \wedge \mathcal{S}(S_i \stackrel{*}{\Rightarrow} x_i))\}\}\} > 0$, where $\mathcal{S}_G(W)$ is the membership grade of the matrix W generated by the 2-FRLG G.

The two dimensional fuzzy regular language (2-FRL) is consisting of the set of all matrices, which are all generated by the 2-FRLG G and is denoted by $L_M(G)$.

Note 1. $\mathcal{S}(y) > 0$ represents the membership grade of $y \in V_h/V_{v_i}$. The grade of membership ρ of H is also represented by ρ/H and the membership value of the derivation chain from α to β is denoted by $\mathcal{S}(\alpha \stackrel{*}{\Rightarrow} \beta)$, where $\alpha,\ \beta$ belongs to either $\{V_h \cup I_h\}^*$ or $\{V_{v_i} \cup T_v\}^*$.

Definition 2.3. [9] A *boustrophedon fuzzy finite automaton* (BFFA) is a sep-tuple $\mathcal{M} = (Q,\ \Sigma,\ \mathcal{R},\ s,\ F,\ \#,\ \Box)$, in which,

- Q represents a collection of finite states.
- Σ represents a collection of input alphabets.
- $s \in Q$ represents fuzzy initial state with $\mathcal{S}(s) \in [0, 1]$ denotes the membership value of the state to be an initial state.
- $F \subseteq Q$ represents a collection of fuzzy final states with $\mathcal{S}(f) \in [0, 1]$; $\forall f \in F$, where $\mathcal{S}(f)$ denotes the membership value of the state to be final
- $\# \notin \Sigma$ represents a special symbol called the border symbol of the rectangular pictures.
- $\Box \notin \Sigma$ represents a new symbol used to represent the erased positions of the picture while scanning.
- $\mathcal{R} : Q \times \Sigma \times Q \to [0, 1]$ represents a finite collection of fuzzy transition rules (the membership value of the rule is 1 if the automata reads the border symbol).

working and configuration of BFFA are explained in [9]

Definition 2.4. [9] A two dimensional word W is said to be accepted by the BFFA $\mathcal{M}$, if the membership value of the word $\mathcal{S}(W) = \{\mathcal{S}(s) \wedge (\vee(\wedge\mathcal{R}^*(s, W, f)) \wedge \mathcal{S}(f)\} > 0$, for some $f \in F$.

Definition 2.5. [9]

$L(\mathcal{M})$ represents the two dimensional fuzzy language accepted by the BFFA $\mathcal{M}$ and is defined by

$$L(\mathcal{M}) = \{W = [a_{ij}]_{m\times n}\ /\ W \text{ is accepted BFFA } \mathcal{M}\}.$$

Definition 2.6. [9] A *returning fuzzy finite automata* (RFFA) is a sep-tuple $\mathcal{M} = (Q, \Sigma, \mathcal{R}, s, F, \#, \Box)$ described same as like in BFFA where as the RFFA always scans the picture from left to right. That is, The RFFA differs from BFFA by its configuration as follows:

Now, a *configuration* (q, W, m), where $q \in Q$, W denotes the two dimensional word and m denotes the current row, is valid only when $1 \leq m \leq |W|_r$ and $\forall i$, $1 \leq i \leq m - 1$, the i^{th} row has entries $\#\Box^{|W|_c-2}\#$, $\forall j$, $m + 1 \leq j \leq |W|_r$, the j^{th} row has entries $\#w\#$, $w \in \Sigma^{|W|_c-2}$ and for some k, $0 \leq k \leq |W|_c - 2, w \in \Sigma^{|W|_c-k-2}$, the m^{th} row equals $\#\Box^k w\#$.

Theorem 2.7. [9] Let Σ be a given alphabet then the language accepted by the BFFA is equivalent to the language accepted by the RFFA. (i.e., L(BFFA)= L(RFFA) over the same alphabet Σ.)

Example 2.8. The fuzzy language $L_\backslash$ consists of $m \times n$ matrices whose main diagonal filled by one and all other entries filled by zeros cannot be accepted by any BFFA.

By using the pumping lemmas and interchanging lemmas stated in [9], it is clear to see that $L_\backslash$ is not accepted any BFFA.

3. Isometric Fuzzy Regular Array Grammars - Boustrophedon Fuzzy Finite Automata relation

In this section, we have introduced isometric fuzzy regular array grammars (IFRAG) and isometric fuzzy regular array languages (IFRAL) generated by IFRAGs. Also, proved the equivalence between IFRAG and BFFA. To study the relationship between IFRAG and BFFA, we first introduced a new notion called direction aware BFFA (dir-BFFA) in the following definition and proved its equivalence with BFFA.

Definition 3.1. A BFFA $\mathcal{M} = (Q, \Sigma, \mathcal{R}, s, F, \#, \Box)$ is said to be a *direction aware BFFA* (dir-BFFA), when there exits a mapping
$\rightarrow : Q \rightarrow \{l, r\}$ such that $\overrightarrow{p} = \overrightarrow{q}$ for every fuzzy transition rule $pa \xrightarrow{\rho} q$ where $a \in \Sigma; \rho \in (0, 1]$ and $\overrightarrow{p} \neq \overrightarrow{q}$ for every fuzzy transition rule $p\# \xrightarrow{\rho} q$ where $\rho \in (0, 1]$. Further, $\overrightarrow{s} = r$.

Lemma 3.2. $L(BFFA) = L(dir - BFFA)$

Proof. Proof of $L(dir - BFFA) \subseteq L(BFFA)$ is trivial, since dir-BFFAs are special cases of BFFAs. The idea behind the proof of $L(BFFA) \subseteq L(dir - BFFA)$ is to track the direction of change in the second part of the state based on whether rows with odd or even numbers are read. $\Box$

3.1. Isometric Fuzzy Regular Array Language.

In this section, we have defined isometric fuzzy regular array grammar and isometric fuzzy regular array language. Also, exemplified the generation of isometric fuzzy regular array languages.

Definition 3.3. An *isometric fuzzy regular array grammar* (IFRAG) G is a 5-tuple $\mathsf{G} = (N, I, P, S, \Box)$, in which

- N represents a collection of finite alphabets called *non-terminals*,
- I represents a collection of finite alphabets called *terminals*,
- $S \in V$ called the *fuzzy start variable* with $\mathcal{S}(S) = \rho \in (0, 1]$ represents the membership value of S to be the fuzzy start variable of G,
- $\Box$ represents the *blank symbol*,
- P represents the collection of *fuzzy production rules* of the form $\Box A \xrightarrow{\rho} Ba$, $A\Box \xrightarrow{\rho} aB$, $\begin{matrix}\Box \\ A\end{matrix} \xrightarrow{\rho} \begin{matrix}B \\ a\end{matrix}$, $\begin{matrix}A \\ \Box\end{matrix} \xrightarrow{\rho} \begin{matrix}a \\ B\end{matrix}$ and $A \xrightarrow{\rho} a$, in which $A, B \in N, a \in I$ and $\rho \in (0, 1]$.

The derivation of an array from an isometric regular array grammar is described below:

- At first, the entire plane is filled with the blank symbols.
- Then, the fuzzy start variable $S \in N$ is placed on some position of the plane with membership value ρ, by replacing $\square$, where $(\rho = \mathcal{S}(S))$.
- On the middle stage, we seen that the non-blank symbols in the plane are all terminal symbols but there exists one non-terminal symbol in the plane say, $A \in N$.
  - The fuzzy production rules of the form $\square A \xrightarrow{\rho} Ba$ can be applied, if the position left of A is blank; in this case, the blank symbol on the left of A is replaced by B and then A is replaced by a with membership value ρ. This type of rule is represented as left fuzzy production rules.
  - The fuzzy production rules of the form $A\square \xrightarrow{\rho} bB$ can be applied in similar way, if the position right of A is blank then. Applying such rule implements a right movement.
  - The fuzzy production rules of the form $\begin{array}{c}\square\\ A\end{array} \xrightarrow{\rho} \begin{array}{c}B\\ a\end{array}$ can be applied, if the position above of A is blank then. This represents an upward movement.
  - The fuzzy production rules of the form $\begin{array}{c}A\\ \square\end{array} \xrightarrow{\rho} \begin{array}{c}a\\ B\end{array}$ can be applied, if the position below of A is blank then. This represents a downward movement.

At any derivation step, the fuzzy production rules of the form $A \xrightarrow{\rho} a$ can be applied and the terminal array is obtained.

Definition 3.4. if $\mathcal{S}(W) = \mathcal{S}(S) \wedge (\vee(S \stackrel{*}{\Rightarrow} W)) > 0$, where $S \in N$, $\mathcal{S}(S) > 0$ and $\mathcal{S}(W)$ is the membership value of the matrix W of size $m \times n$ generated by the IFRAG G then the matrix W is generated by the IFRAG G.

The *isometric fuzzy regular array language* (IFRAL) generated by a IFRAG G is represented by $L(\mathsf{G})$ and is defined by

$$L(\mathsf{G}) = \{W = [a_{ij}]_{m\times n}/W \text{ is generated by IFRAG } \mathsf{G}\}.$$

Notation: $L_{rec}(\mathsf{G})$ denotes the collection of all two dimensional words generated by IFRAG G, in which all the entries of any two dimensional word does not contain a blank symbol $\square$. That is, the language consists of rectangular arrays whose all entries are filled with symbols taken from a finite alphabet.

Example 3.5. Consider a IFRAG $\mathsf{G} = (N, I, P, S, \square)$, where $N = \{S, A, B, E, F\}$, $I = \{Z, *\}$, $S \in N$ represents the fuzzy start variable with

$\mathcal{S}(S) = 0.8$, $\Box$ - blank symbol and

$$P = \{ S\Box \xrightarrow{0.3} ZA, A\Box \xrightarrow{0.4} *A, \Box B \xrightarrow{0.6} B*, E\Box \xrightarrow{0.2} ZE, \Box F \xrightarrow{0.6} FZ,$$
$$\begin{matrix} A \\ \Box \end{matrix} \xrightarrow{0.8} \begin{matrix} * \\ B \end{matrix}, \begin{matrix} A \\ \Box \end{matrix} \xrightarrow{0.9} \begin{matrix} * \\ F \end{matrix}, \begin{matrix} B \\ \Box \end{matrix} \xrightarrow{0.6} \begin{matrix} Z \\ E \end{matrix}, E \xrightarrow{0.8} Z, F \xrightarrow{0.9} Z \}.$$

The IFRAL $L(\mathsf{G})$ consists of $m \times n$ matrices describing L token with membership value 0.3, for $m = 2,\ n = 2$ and 0.2 for $m,\ n \geq 3$.

The generation of L token of size 2×2 matrix W by IFRAG G is described below:

$$\begin{matrix} \Box & \Box \\ \Box & \Box \end{matrix} \xRightarrow{0.8} \begin{matrix} S & \Box \\ \Box & \Box \end{matrix} \xRightarrow{0.3} \begin{matrix} Z & A \\ \Box & \Box \end{matrix} \xRightarrow{0.9} \begin{matrix} Z & * \\ \Box & F \end{matrix} \xRightarrow{0.6} \begin{matrix} Z & * \\ F & * \end{matrix} \xRightarrow{0.9} \begin{matrix} Z & * \\ Z & Z \end{matrix}$$

The degree of derivabilty of W describing token L with $\mathcal{S}(S) = 0.8$ is $\mathcal{S}(S) \wedge (\vee(\mathcal{S}(S \overset{*}{\Rightarrow} X))) = 0.8 \wedge (\vee(0.3 \vee 0.9 \vee 0.6 \vee 0.9)) = 0.8 \wedge 0.3 = 0.3$.

Hence, the 2×2 matrix describing token L with membership value 0.3 generated by IFRAG is in $L(\mathsf{G})$.

Example 3.6. Consider the IFRAG $\mathsf{G} = (N, I, P, S, \Box)$, where $N = \{S, A, B, C, D, E, F\}$, $I = \{0, 1\}$, S is the fuzzy start variable with $\mathcal{S}(S) = 0.9$, $\Box$ is the blank symbol and

$$P = \{ \begin{matrix} \Box \\ S \end{matrix} \xrightarrow{0.8} \begin{matrix} A \\ 1 \end{matrix}, \begin{matrix} \Box \\ A \end{matrix} \xrightarrow{0.7} \begin{matrix} A \\ 0 \end{matrix}, \begin{matrix} B \\ \Box \end{matrix} \xrightarrow{0.8} \begin{matrix} 1 \\ F \end{matrix}, \begin{matrix} C \\ \Box \end{matrix} \xrightarrow{0.6} \begin{matrix} 0 \\ C \end{matrix}, \begin{matrix} \Box \\ H \end{matrix} \xrightarrow{0.6} \begin{matrix} E \\ 0 \end{matrix},$$
$$\begin{matrix} \Box \\ E \end{matrix} \xrightarrow{0.5} \begin{matrix} A \\ 1 \end{matrix}, \begin{matrix} F \\ \Box \end{matrix} \xrightarrow{0.6} \begin{matrix} 0 \\ C \end{matrix}, \Box A \xrightarrow{0.6} B0, \Box B \xrightarrow{0.6} B0, C\Box \xrightarrow{0.5} OH,$$
$$H\Box \xrightarrow{0.8} 0H, S \xrightarrow{0.7} 1, F\Box \xrightarrow{0.6} 0, E \xrightarrow{0.7} 1 \}.$$

The language generated by IFRAG is $L_{\setminus}$ consists of $m \times n, m, n = 1, 2, \ldots$ whose main diagonal is filled by one and all other entries are filled by zero with membership value 0.7 when $m = n = 1$, 0.6 when $m = n = 2$ and $m = n \leq 3$.

Definition 3.7. If the IFRAG G has no left (right/ upwards / downwards) movements then it is represented by $\overline{L} - IFRAG$ ($\overline{R} - IFRAG$ / $\overline{U} - IFRAG$ / $\overline{D} - IFRAG$, respectively). Let $X \in \{U, D, L, R\}$. Then,
$L(\overline{X} - IFRAG) = \{L(\mathsf{G}) : \mathsf{G}$ is a $\overline{X} - IFRAG$ over $I\}$.
$L_{rec}(\overline{X} - IFRAG) = \{L_{rec}(\mathsf{G}) : \mathsf{G}$ is an $X - IFRAG$ over $I\}$.

3.2. Closure properties of Isometric Fuzzy Regular Array Languages. In this section, we have defined and discussed the closure properties of IFRAL such as reflection about the right-most vertical, reflection about the base (horizontal) and transpose

Definition 3.8. If W is a matrix as given below, then the *reflection about the right-most vertical*, *reflection about the base* and *transpose* are represented by

$R_v(W)$, $R_h(W)$ and $T(W)$ respectively and they are defined as follows.

$$W = \begin{matrix} u_{11} & \dots & u_{1n} \\ \dots & \dots & \dots \\ u_{m1} & \dots & u_{mn} \end{matrix} \qquad R_v(W) = \begin{matrix} u_{1n} & \dots & u_{11} \\ \dots & \dots & \dots \\ u_{mn} & \dots & u_{m1} \end{matrix}$$

$$R_h(W) = \begin{matrix} u_{m1} & \dots & u_{mn} \\ \dots & \dots & \dots \\ u_{11} & \dots & u_{1n} \end{matrix} \qquad T(W) = \begin{matrix} u_{11} & \dots & u_{m1} \\ \dots & \dots & \dots \\ u_{1n} & \dots & u_{mn} \end{matrix}$$

The transpose of a IFRAL L_{rec} is denoted by $T(L_{rec})$ and is defined by $T(L_{rec}) = \{T(W)/W \in L_{rec}\}$. The reflection about the right-most vertical and reflection about the base of the IFRAL are defined in the similar way.

Theorem 3.9. *The isometric fuzzy regular array languages are closed under the operations reflection about the right-most vertical, reflection about the base and transpose.*

Proof. The closure of reflection about vertical of IFRAL is described below.

Let L_{rec} be the isometric fuzzy regular array language generated by IFRAG $\mathsf{G} = (N, I, P, S, \square)$. Construct another IFRAG
$\mathsf{G}' = (N', I', P', S', \square)$ such that $N' = N$, $I' = I$, S' is fuzzy start variable corresponding to fuzzy start variable S with $\mathcal{S}(S') = \mathcal{S}(S)$, $\square$ is blank symbol and P' of G' is defined as follows:

All the fuzzy production rules in P of the form $\begin{matrix} \square \\ A \end{matrix} \xrightarrow{\rho} \begin{matrix} B \\ a \end{matrix}$, $\begin{matrix} A \\ \square \end{matrix} \xrightarrow{\rho} \begin{matrix} a \\ B \end{matrix}$ and $A \xrightarrow{\rho} a$, where $A, B \in N, a \in I$ and $\rho \in (0, 1]$ are defined in P' exactly as in P. For each fuzzy production rule of the form $\square A \xrightarrow{\rho} Ba; A, B \in N$, $a \in I$ and $\rho \in (0, 1]$ in P add the fuzzy production rule $A\square \xrightarrow{\rho} aB; A, B \in N'$, $a \in I'$ and $\rho \in (0, 1]$ in P' and for each fuzzy production rule of the form $A\square \xrightarrow{\rho} aB; A, B \in N, a \in I$ and $\rho \in (0, 1]$ in P add the fuzzy production rule $\square A \xrightarrow{\rho} Ba; A, B \in N', a \in I'$ and $\rho \in (0, 1]$ in P'.

By the above construction of G', it is clear to see that, if $L_{rec}(\mathsf{G})$ is a IFRAL generated by IFRAG G then $L_{rec}(\mathsf{G}')$ is also a IFRAL, which is the reflection about vertical of the IFRAL $L_{rec}(\mathsf{G})$.

Hence, if L_{rec} is a isometric fuzzy regular array language then its reflection about vertical $R_v(L_{rec})$ is also a isometric fuzzy regular array language.

i.e., The class of isometric fuzzy regular array languages are closed under reflection about the vertical.

The other closure properties can be similarly proved. $\square$

Corollary 3.10.

- $R_v(L_{rec}(\overline{R} - IFRAG)) = L_{rec}(\overline{L} - IFRAG)$,
  $R_v(L_{rec}(\overline{L} - IFRAG)) = L_{rec}(\overline{R} - IFRAG)$,
- $R_h(L_{rec}(\overline{U} - IFRAG)) = L_{rec}(\overline{D} - IFRAG)$,
  $R_h(L_{rec}(\overline{D} - IFRAG)) = L_{rec}(\overline{U} - IFRAG)$,

- $T(L_{rec}(\overline{L} - IFRAG)) = L_{rec}(\overline{U} - IFRAG)$,
  $T(L_{rec}(\overline{U} - IFRAG)) = L_{rec}(\overline{L} - IFRAG)$,
  $T(L_{rec}(\overline{D} - IFRAG)) = L_{rec}(\overline{R} - IFRAG)$,
  $T(L_{rec}(\overline{R} - IFRAG)) = L_{rec}(\overline{D} - IFRAG)$.

3.3. Relation between IFRAL and BFFA. In this section, we have discussed the relation between the language generated by isometric fuzzy regular array grammars and the language accepted by boustrophedon fuzzy finite automata.

Theorem 3.11. $L(BFFA) = L_{rec}(\overline{U} - IFRAG)$.

Proof. Let $\mathcal{M} = (Q, \Sigma, \mathcal{R}, (s, r), F, \#, \square)$ be a dir-BFFA. We can construct a IFRAG $\mathsf{G} = (N, I, P, S, \square)$ such that

- $N = Q \times \{l, r\}$ where l and r represents the left and right direction of movement of the BFFA.
- $I = \Sigma$.
- S is the fuzzy start variable corresponds to the fuzzy start state (s, r) with $\mathcal{S}(S) = \mathcal{S}(s, r)$
- The fuzzy production rules P of G is obtained from the fuzzy transitions $\mathcal{R}$ of $\mathcal{M}$ as follows:
  - $\square(q, l) \xrightarrow{\rho} (p, l)a$ is in P for every $(q, l)a \xrightarrow{\rho} (p, l)$ in $\mathcal{R}$ and $(p, r)\square \xrightarrow{\rho} a(q, r)$ is in P for every $(p, r)a \xrightarrow{\rho} (q, r)$ in $\mathcal{R}$.
  - $(q, l) \xrightarrow{\rho} a$ or $(q, r) \xrightarrow{\rho} a$ if (q, l) or $(q, r) \in F$ with $\mathcal{S}(q, l) = \mathcal{S}(q, r) = \rho$
  - $\begin{matrix}(p, r)\\ \square\end{matrix} \xrightarrow{\rho} \begin{matrix}a\\ (q', l)\end{matrix}$ is in P when $(p, r)a \xrightarrow{\rho'} (q, r)$ and $(q, r)\square \xrightarrow{\rho''} (q', l)$ is in $\mathcal{R}$, where $\rho = \min\{\rho', \rho''\}$.
    Also, $\begin{matrix}(p, l)\\ \square\end{matrix} \xrightarrow{\rho} \begin{matrix}a\\ (q', r)\end{matrix}$ is in P when $(p, l)a \xrightarrow{\rho'} (q, l)$ and $(q, l)\square \xrightarrow{\rho''} (q', r)$ is in $\mathcal{R}$, where $\rho = \min\{\rho', \rho''\}$

It is easy to observe that, for every $m \times n$ matrix W generated by an $\overline{U} - IFRAG$ there exits a BFFA that accepts W.

Therefore, for every $L(\mathsf{G})$ generated by the $\overline{U} - IFRAG$ G there exists a BFFA M that accepts the $L(\mathsf{G})$.

The converse part can be similarly proved.

Hence, $L(BFFA) = L_{rec}(\overline{U} - IFRAG)$. $\square$

Corollary 3.12. $T(L(BFFA)) = L_{rec}(\overline{L} - IFRAG)$.

The proof of the corollary is easily verifiable from Theorem 3.11 and Corollary 3.10.

4. Two dimensional fuzzy regular langauges - Boustrophedon Fuzzy Finite Automata relation

In this section, we have discussed the relation between two dimensional fuzzy right - linear grammar with returning fuzzy finite automata and boustrophedon fuzzy finite automata.

Theorem 4.1. *For every 2-FRLG G there exists a RFFA $\mathcal{M}$ such that the transpose of the language generated by the 2-FRLG is equal to the language accepted by the RFFA.*

Proof. Let $G = (G_h, G_v)$ be a 2-FRLG, where $G_h = (V_h, I_h, R_h, S)$ and $G_v = \bigcup_{i=1}^{n} G_{v_i} = (V_{v_i}, I_v, P_{v_i}, S_i)$. We can construct an RFFA $\mathcal{M} = (Q, \Sigma, \mathcal{R}, s, F, \#, \Box)$ such that $Q = (V_h \times \{f\}) \times (V_v \times \{f\}) \cup \{s\}$, where $f \notin V_h \cup V_v$ and s is the fuzzy start state of RFFA corresponding to the start variable $S \in G_h$ with $\mathcal{S}(s) = \mathcal{S}(S)$, $\Sigma = I_v$, $F = \{(f, f)\}$ with $\mathcal{S}(f, f) = 1$ and the fuzzy transition rules $\mathcal{R}$ are defined as follows:

- $sa \xrightarrow{\rho} (S', A')$ is in $\mathcal{R}$ when $S \xrightarrow{\rho'} AS' \in R_h$ and $A \xrightarrow{\rho''} aA' \in R_v$ in which $\rho = \min\{\rho', \rho'', \rho'''\}$ where $\mathcal{R}(S) = \rho'''$.
- $s \xrightarrow{\rho} (f, A')$ is in $\mathcal{R}$ when $S \xrightarrow{\rho'} A \in R_h$ and $A \xrightarrow{\rho''} aA' \in R_v$ in which $\rho = \min\{\rho', \rho'', \rho'''\}$ where $\mathcal{R}(S) = \rho'''$.
- $(Y, A)a \xrightarrow{\rho} (Y, A')$ is in $\mathcal{R}$ when $Y \in V_h \cup \{f\}$ and $A \xrightarrow{\rho'} aA' \in R_v$ in which $\rho = \min\{\rho', \rho''; \rho'' = \mathcal{R}(S)\}$ otherwise $\rho = \rho'$.
- $(Y, A)a \xrightarrow{\rho} (Y, f)$ is in $\mathcal{R}$ when $A \xrightarrow{\rho'} a \in R_v$ and $Y \in V_h$ in which $\rho = \min\{\rho', \rho''; \rho'' = (S)\}$ otherwise $\rho = \rho'$.
- $(Y, f)\# \xrightarrow{\rho} (Y', A)$ is in $\mathcal{R}$ when $Y \xrightarrow{\rho} AY' \in R_h$.
- $(Y, f)\# \xrightarrow{\rho} (f, A)$ is in $\mathcal{R}$ when $Y \to \rho A \in R_h$.
- $(f, A)a \xrightarrow{\rho} (f, f)$ is in $\mathcal{R}$ when $A \xrightarrow{\rho} a \in R_v$.

where ρ, ρ' and $\rho''' \in [0, 1]$.

It is easy to observe that for every $m \times n$ matrix W in the language generated by 2-FRLG then there exists a RFFA $\mathcal{R}$, that accepts the transpose of the matrix W.

Hence, for every 2-FRLG there exists a RFFA and that accepts the transpose of 2-FRL generated by 2-FRLG. □

Theorem 4.2. *For every RFFA $\mathcal{M}$ there exists a 2-FRLG G such that the $2-FRL$ accepted by the RFFA is equal to the transpose of the language generated by 2-FRLG.*

Proof. Let $\mathcal{M} = (Q, \Sigma, \mathcal{R}, s, F, \#, \Box)$ be an RFFA. We can construct a 2-FRLG $G = (G_h, G_v)$ where $G_h = (V_h, I_h, R_h, S)$ and $G_v = \bigcup_{i=1}^{n} G_{v_i} = (V_{v_i}, I_v, P_{v_i}, S_i)$ such that

- $V_h = Q \cup \{S\}$, $V_v = I_h$, $I_v = \Sigma$ and S is the start symbol with membership value of start state of RFFA $\mathcal{M}$
- R_h is defined as follows:
  - $S \xrightarrow{\rho} (S,p)p \in R_h$ for each $p \in Q$, where $\rho = \mathcal{S}(S)$.
  - $p_1 \xrightarrow{\rho} (p_1,p_2)p_2 \in R_h$ for all $p_1, p_2 \in Q$ where $\rho = \mathcal{S}(S)$ if $p_1, p_2 = S$ otherwise $\rho = 1$
  - $p_1 \xrightarrow{\rho} (p_1, f) \in R_h$ for each $f \in F, p_1 \in Q$ where $\rho = \mathcal{S}(f) \forall f \in F$
- R_v is defined as follows:
  - $(p_1,p_3) \xrightarrow{\rho} a(p_2,p_3) \in R_v$ for each $p_1 a \xrightarrow{\rho'} p_2 \in \mathcal{R}$ and $p_3 \in Q$, in which $\rho = \min\{\rho', \rho''\}$ where $\rho'' = \mathcal{S}(S)$ otherwise $\rho = \rho'$
  - $(p_1,p_3) \xrightarrow{\rho} a \in R_v$ for each $p_1 a \xrightarrow{\rho} p_2 \in \mathcal{R}$ and $p_2\# \xrightarrow{1} p_3 \in \mathcal{R}$
  - $(p_1, f) \xrightarrow{\rho} a \in R_v$ for each $p_1 a \xrightarrow{\rho'} f \in \mathcal{R}$ and $f \in F$, in which $\rho = \min\{\rho', \rho''\}$ where $\rho'' = d(f)$

It is easy to observe that for every $m \times n$ matrix W in the language accepted by the RFFA $\mathcal{M}$ there exists a 2-FRLG G that generates the transpose of matrix W.

Hence, for every RFFA there exists a 2-FRLG such that the fuzzy language accepted by the RFFA is equal to the transpose of the language generated by 2-FRLG. □

Corollary 4.3. *L(RFFA)=T(L(2-FRLG)).*

Proof of Corollary 4.3 can be easily obtained from the Theorems 4.1 and 4.2.

Example 4.4. Consider the 2-FRLG $G = (G_h, G_v)$ generates 2-FRL $L(G)$ describing token L of size $m \times n$ with degree 0.2, when $m > 1, n = 2$ and 0.1, when $m > 1, n > 1$ as given in Example 3.1 in Section 3 of [8]. Here, $G_h = (\{S, A\}, \{S \xrightarrow{0.2} S_1 A, A \xrightarrow{0.1} S_2 A, A \xrightarrow{0.3} S_2\}, \{1/S\})$

$G_v = G_{v_1} \cup G_{v_2}$

$G_{v_1} = (\{S_1\}, \{Z, *\}, \{S_1 \xrightarrow{0.3} Z S_1, S_1 \xrightarrow{0.6} Z\}, \{1/S_1\})$

$G_{v_2} = (\{S_2, B\}, \{Z, *\}, \{S_2 \xrightarrow{0.2} *B, B \xrightarrow{0.2} *B, B \xrightarrow{0.3} Z\}, \{1/S_2\})$

By Corollary 3.1, we can construct a RFFA $\mathcal{M} = (Q, \Sigma, \mathcal{R}, s, F, \#, \Box)$, where

$$Q = \{s, (A,S_1), (A,S_2), (A,B), (S,S_1), (S,B), (S,S_2), (f,S_1), (S,f), (A,f), (f,S_2), (f,B), (f,f)\}$$

$\Sigma = I_v$, s is the start state corresponds to the start variable S with $\mathcal{S}(s) = \mathcal{S}(S)$, $F = \{1/(f,f)\}$, $\#$ represents the blank symbol and the transition rules $\mathcal{R}$ of RFFA is defined as follows:

- For $S \xrightarrow{0.2} S_1 A \in R_h$ and $S_1 \xrightarrow{0.3} Z S_1 \in R_v$, we have $sZ \xrightarrow{0.2} Z S_1 \in \mathcal{R}$.
- For each $V_h \cup \{f\}$ and $S_1 \xrightarrow{0.3} Z S_1$, $S_2 \xrightarrow{0.2} *B$, $B \xrightarrow{0.2} *B$, we have the following transition rules in $\mathcal{R}$.

- $(S,S_1)Z \xrightarrow{0.3} (S,S_1)$
  $(S,S_2)* \xrightarrow{0.2} (S,B)$
  $(S,B)* \xrightarrow{0.2} (S,B)$
- $(A,S_1)Z \xrightarrow{0.3} (A,S_1)$
  $(A,S_2)* \xrightarrow{0.2} (A,B)$
  $(A,B)* \xrightarrow{0.2} (A,B)$
- $(f,S_1)Z \xrightarrow{0.3} (f,S_1)$
  $(f,S_2)* \xrightarrow{0.2} (f,B)$
  $(f,B)* \xrightarrow{0.2} (f,B)$

- For each $V_h = \{S, A\}$ and $S_1 \xrightarrow{0.6} Z$, $S_1 \xrightarrow{0.6} Z$, $B \xrightarrow{0.3} Z$ in R_v, we have the following transition rules in $\mathcal{R}$.

  - $(S,S_1)Z \xrightarrow{0.6} (S,f)$
    $(A,S_1)Z \xrightarrow{0.6} (A,f_1)$
  - $(S,B)Z \xrightarrow{0.3} (S,f)$
    $(A,B)Z \xrightarrow{0.3} (A,f)$

- For $S \xrightarrow{0.2} S_1A$ and $A \xrightarrow{0.1} S_2A$ in R_h, we have $(S,f)\# \xrightarrow{0.2} (A,S_1)$ and $(A,f)\# \xrightarrow{0.1} (A,S_2)$ in $\mathcal{R}$.
- For $A \xrightarrow{0.3} S_2$ in R_h, we have $(A,f)\# \xrightarrow{0.3} (f,S_2)$ in $\mathcal{R}$.
- For $S_1 \xrightarrow{0.6} Z$ and $B \xrightarrow{0.3} Z$ in R_v, we have $(f,B)Z \xrightarrow{0.3} (f,f)$ in $\mathcal{R}$.

Corollary 4.5. *L(BFFA)=T(L(FRLG)).*

Proof is obvious from Theorem 1 in Section 3 of [9] and Corollary 4.3.

Remark 4.1. Here we consider that 2-FRLG G has only one fuzzy start variable.

Remark 4.2. The RFFA and BFFA constructed from 2-FRLG can read the # symbol with membership value $\rho \in (0,1]$.

By Theorem 4.1, Theorem 4.2, Corollary 3.10, Corollary 4.3 and Corollary 4.5, we obtain the following results.

Corollary 4.6. $T(L(BFFA)) = L_{rec}(\overline{L} - IFRAG)$.

Corollary 4.7.
- $L_{rec}(\overline{R} - IFRAG) = L_{rec}(\overline{L} - IRAG) = L(2 - FRLG)$.
- $L_{rec}(\overline{D} - IFRAG) = L_{rec}(\overline{U} - IFRAG) = L(BFFA)$

Lemma 4.8. *The two dimensional fuzzy language L_{-} of $m \times n$ matrices with any one row completely filled by one and zeros on all other entries cannot be generated by any 2-FRLG.*

Proof. Consider that there exists an 2-FRLG G that generates $L_{|}$. Since $T(L_{-}) = L_{|}$ and by Corollary 4.2, there exists a BFFA that accepts $T(L_{|})$, which is a contradiction to our result that $L_{|}$ cannot be accepted by any BFFA as stated in pumping lemma of Section 4 of [9].

Therefore, the two dimensional fuzzy language L_{-} cannot be generated by any 2-FRLG. □

Lemma 4.9. *The two dimensional fuzzy language L of $n \times n$ matrices with ones on the main diagonal and zeros on all other entries cannot be generated by any 2-FRLG.*

Proof. Let us assume that there exists an 2-FRLG G that generates $L_\backslash$. Since $T(L_\backslash) = L_\backslash$ and by Corollary 4.5, there exists a BFFA that accepts $T(L_\backslash)$, which is a contradiction to our result that $L_\backslash$ cannot be accepted by any BFFA as stated in Section 4 of [9].

Therefore, the two dimensional fuzzy language $L_\backslash$ cannot be generated by any 2-FRLG. □

5. Hierarchy Results

In this section, the hierarchy results of the two dimensional fuzzy languages generated by IFRAG, 2-FRLG and the two dimensional fuzzy languages accepted by BFFA has been discussed and illustrated.

Theorem 5.1. $L(BFFA) \cup L(2-FRLG) \subset L_{rec}(IFRAG)$.

Proof. It is easy to observe that the proof of $L(BFFA) \cup L(2-FRLG) \subseteq L_{rec}(IFRAG)$ is trivial, since by Corollary 3.10 and Theorem 3.11. The proof of $L(BFFA) \cup L(2-FRLG) \neq L_{rec}(IFRAG)$ can be obtained from Example 3.6 and interchanging lemma stated in [9]. □

The proper inclusion and incomparabilities illustrated in Figure 1 can be easily seen from Theorem 5.1, Corollary 3.10, Example 3.6, Lemma 4.8 and Lemma 4.9. Since, $L(2-FRLG)$ and $L(BFFA)$ are incomparable the proper inclusion $L(2-FRLG) \cap L(BFFA)$ is shown in Figure 1 can be easily verifiable.

Note 2. $L(BFFA) \cap L(2-FRLG) \neq \emptyset$, since the two dimensional fuzzy languages describing token $'L'$ is in $L(BFFA)$ and $L(2-FRLG)$ (see Example 3.1 in [8] and Example 1 in [9]).

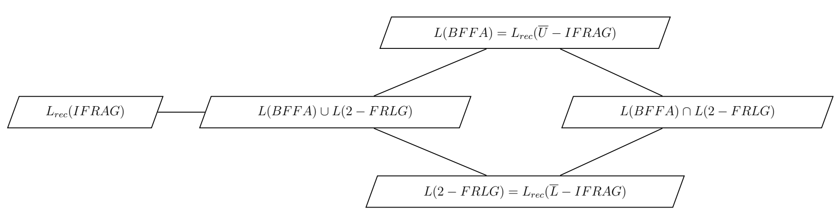

Figure 1. Relation among the two dimensional fuzzy languages

6. Conclusion

In this paper, one of the simplest model to generate two dimensional fuzzy regular languages called isometric fuzzy regular array grammars has been introduced. Also, exemplified the generation of isometric fuzzy regular array languages. Proved the closure properties of isometric fuzzy regular array languages. The relation between isometric fuzzy regular array grammars with boustrophedon fuzzy finite automata has been shown. Further, discussed the relations between two dimensional fuzzy regular grammars with returning fuzzy finite automata and boustrophedon fuzzy finite automata. An example is also provided. Also, studied the hierarchy results of two dimensional fuzzy regular languages, isometric fuzzy regular array languages and the languages accepted by boustrophedon fuzzy finite autoamata. The future work of this study is to extend the proposed isometric fuzzy regular array languages in line with isometric fuzzy context-free array languages and to study their application in generating various kolam and tiling patterns.

Conflicts of interest : The authors declare no conflict of interest.

Data availability : Not applicable

References

1. P.R. Asveld, *Fuzzy context-free languages-part 2: Recognition and parsing algorithms*, Theoretical computer science **347** (2005), 191–213.
2. P.R.J. Asveld, *Fuzzy context-free languages-part 1: Generalized fuzzy context-free grammars*, Theoretical Computer Science **347** (2005), 167–190.
3. C.R. Cook and P. Shen-Pei Wang, *A chomsky hierarchy of isotonic array grammars and languages*, Computer Graphics and Image Processing **8** (1978), 144–152.
4. H. Fernau, M. Paramasivan, M.L. Schmid and D.G. Thomas, *Simple picture processing based on finite automata and regular grammars*, Journal of Computer and System Sciences **95** (2018), 232–258.
5. D. Giammarresi and A. Restivo, *Two-dimensional languages*, In Handbook of Formal Languages: Volume 3 Beyond Words, Springer, Berlin, Heidelberg, 1997, 215–267.
6. J.E. Hopcroft, R. Motwani and J.D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, J. Pearson Education Ltd., 3rd edition, 2014.
7. K. Inoue and I. Takanami, *A survey of two-dimensional automata theory*, In International Meeting of Young Computer Scientists Springer, 1988, 72–91.
8. A. John Kaspar, D.K. Sheena Christy, V. Masilamani and D. Thomas, *Two dimensional fuzzy regular languages*, Fuzzy Sets and Systems **422** (2022), 309–330.
9. A. John Kaspar, D.K. Sheena Christy and D.G. Thomas, *Two diemnsional picture processing using broustropheodon fuzzy finite automata*, AIP Conference Proceedings, 2022, In press.
10. G. Klir and B. Yuan, *Fuzzy sets and fuzzy logic*, volume 4. Prentice hall, New Jersey, 1995.
11. Y. 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.
12. P. Linz, *An introduction to formal languages and automata*, Jones & Bartlett Learning, 2006.

13. J.N. Mordeson and D.S. Malik, *Fuzzy automata and languages: theory and applications*, CRC Press, 2002.
14. A. Rosenfeld, *Isotonic grammars, parallel grammars and picture grammars*, Technical report, Maryland university college park computer science center, 1970.
15. A. Rosenfeld and R. Siromoney, *Picture languages—a survey*, Languages of design **1** (1993), 229–245.
16. S. Sharan, B.K. Sharma and K. Jacob, *Interval type-2 fuzzy automata and interval type-2 fuzzy grammar*, Journal of Applied Mathematics and Computing **68** (2022), 1505-1526.
17. G. Siromoney, R. Siromoney and K. Krithivasan, *Picture languages with array rewriting rules*, Information and Control **22** (1973), 447–470.
18. G. Siromoney, R. Siromoney and K. Krithivasan, *Array grammars and kolam*, Computer Graphics and Image Processing **3** (1974), 63–82.
19. R. Siromoney, *Advances in array languages*, In International Workshop on Graph Grammars and Their Application to Computer Science, Springer, 1986, 549–563.
20. T.J. Smith, *Two-dimensional automata*, Technical report, Queen's University, Kingston, 2019, 2019–637.
21. G. Stromoney, R. Siromoney, and K. Krithivasan, *Abstract families of matrices and picture languages*, Computer Graphics and Image Processing **1** (1972), 284–307.
22. P.S-P. Wang, *Some new results on isotonic array grammars*, Information Processing Letters **10** (1980), 129–131.
23. W.G. Wee and K.S. Fu, *A formulation of fuzzy automata and its application as a model of learning systems*, Readings in Fuzzy Sets for Intelligent Systems, 1993, 892–900.
24. Q. Wu, X. Wang, Z. Chen, H. Chen, D. Sun, and Z. Han, *Application of fuzzy pushdown automaton on prediction of quality control for spinning yarn: Application of fpda on quality prediction*, Information Technology and Control **1** (2021), 76–88.
25. L.A. Zadeh, *Fuzzy sets*, Information and Control **8** (1965), 338–353.

A. John Kaspar received Ph.D from SRM Institute of Science and Technology in 2023 and is currently working as an Assistant Professor in Department of Mathematics, CHRIST (Deemed to be University). His research interests include formal languages, automata, fuzzy languages, fuzzy automata and picture languages.

Department of Mathematics, CHRIST (Deemed to be University), Bengaluru - 560029, Karnataka, India.
e-mail: johnkaspara@gmail.com

D.K. Sheena Christy received M.Sc., M. Phil and Ph.D. in Mathematics from University of Madras in 2001, 2004 and 2016 respectively. Since 2011 she has been working as an Assistant Professor in Department of Mathematics, Faculty of Engineering and Technology, SRM Institute of Science and Technology. Her research interests are formal languages, automata, fuzzy languages, fuzzy automata and splicing systems on picture languages.

Department of Mathematics, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu-603 203, Tamilnadu India.
e-mail: sheena.lesley@gmail.com

D.G. Thomas received the M.Sc. degree in mathematics in 1981 from the Madurai Kamaraj University and the M.Phil and the Ph.D. degrees in 1984 and 1996, respectively from Madras University. He is a former Professor in Department of Mathematics, Madras Christian College. His areas of interest include formal languages, automata, string rewriting systems, infinite and bi-infinite words, codes, learning, cryptography and picture languages.

Department of Mathematics, Madras Christian College, Tambaram, Chennai-600 059, India.
e-mail: dgthomasmcc@yahoo.com