

CERTAIN COMPARISON RESULTS ON PURE TWO DIMENSIONAL CONTEXT FREE GRAMMARS AND ITS VARIANTS

Submitted in partial fulfillment of the requirements for the award of
Masters of Science Degree in Mathematics

By

R. HARITA (40630003)



**DEPARTMENT OF MATHEMATICS
SCHOOL OF SCIENCE AND HUMANITIES**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119**

MAY - 2022



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF MATHEMATICS

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **R HARITA (40630003)**. Who carried out the project entitled "**CERTAIN COMPARISON RESULTS ON PURE TWO DIMENSIONAL CONTEXT FREE GRAMMARS AND ITS VARIANTS**" under our supervision from **January 2022 to May 2022**.

(DR. T. KAMARAJ)
Internal Guide
(Name in Capital letters
with Signature)

External Guide (if Applicable)
(Name in Capital letters
with signature)

(DR. M. NIRMALA)
Head of the Department
Dept. of Mathematics
Sathyabama Institute of Science and Technology
(Deemed to be University), Chennai-600 119.

(Name in Capital letters with Seal & Signature)

Submitted for Viva voce Examination held on 12.05.2022

Internal Examiner
(Name in Capital letters
with Signature)
(Dr. R. Manoharan)

External Examiner
(Name in Capital letters
with signature)
(S. GANESH)

DECLARATION

I R HARITA (40630003) hereby declare that the Project Report entitled "CERTAIN COMPARISON RESULTS ON PURE TWO DIMENSIONAL CONTEXT FREE GRAMMARS AND ITS VARIANTS" done by me under the guidance of Dr. T. KAMARAJ (Internal) and _____ (External) at SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY submitted in partial fulfillment of the requirements for the award of Masters of Science Degree in Mathematics.

DATE: 02/05/2022

PLACE: CHENNAI



SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to the Board of Management of SATHYABAMA, for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to Dr. M. NIRMALA, Head of the Department, Dept. of Mathematics for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Dr. T. KAMARAJ, Assistant Professor, Department of Mathematics, for his valuable guidance, suggestions, and constant encouragement that paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Mathematics who were helpful in many ways for the completion of the project.

ABSTRACT

In the study of two-dimensional languages, in the recent past, pure 2D context-free grammars and their variants were introduced which make use of only terminal symbols in their production rules and allow rewriting of any (specific) row or column of the rectangular array. We draw some comparisons on the expressiveness of these grammars with other interesting rectangular array grammars like Prusa Grammars, Parallel contextual array insertion and deletion grammars, and Regional Tile rewriting grammars which make use of terminals as well as non-terminal symbols in their derivation rules.

KEY WORDS: Pure 2D context-free grammars, Prusa Grammars, Parallel contextual array insertion and deletion grammars, Regional tile rewriting grammars

CONTENT

ABSTRACT	(i)
LIST OF ABBREVIATIONS	(iii)

CHAPTERS

I. INTRODUCTION	2
1.1 PURE 2D CONTEXT-FREE GRAMMAR	3
1.2 ($/ / u$) PURE 2D CONTEXT FREE GRAMMAR	5
1.3 (r/d) PURE 2D CONTEXT FREE GRAMMAR	6
1.4 PRUSA GRAMMAR	7
1.5 REGIONAL TILE REWRITING GRAMMAR	8
1.6 PARALLEL CONTEXTUAL ARRAY INSERTION AND DELETION GRAMMAR	9
II. LITERATURE SURVEY	13
III. COMPARISON OF PURE 2D CONTEXT-FREE GRAMMARS WITH PRUSA GRAMMARS AND PARALLEL CONTEXTUAL ARRAY INSERTION AND DELETION GRAMMARS	15
IV. COMPARISON OF VARIANTS OF PURE 2D CONTEXT-FREE GRAMMARS WITH PRUSA AND REGIONAL TILE REWRITING GRAMMARS	23
V. CONCLUSION	29
VI. REFERENCES	30

LIST OF ABBREVIATION

1. P2DCFG – PURE TWO DIMENSIONAL CONTEXT-FREE GRAMMAR
2. (l/u) P2DCFG – (LOWERMOST/UPPERMOST) PURE TWO DIMENSIONAL CONTEXT FREE GRAMMAR
3. (r/d) P2DCFG – (RIGHTMOST/DOWNMOST) PURE TWO DIMENSIONAL CONTEXT FREE GRAMMAR
4. PG – PRUSA GRAMMAR
5. RTG – REGIONAL TILE REWRITING GRAMMAR
6. PCAIDG – PARALLEL CONTEXTUAL ARRAY INSERTION & DELETION GRAMMAR
7. Σ^{**} - THE SET OF ALL RECTANGULAR ARRAYS OVER Σ
8. Σ^* - THE SET OF ALL STRINGS OVER Σ

CHAPTER - 1

INTRODUCTION

Syntactic techniques of the generation of digital picture arrays have emerged as one of the major areas of theoretical studies in picture analysis, basically due to the structure handling ability of the syntactic models. Several two-dimensional (2D) rectangular and non-rectangular picture generating or recognizing mechanisms such as two-dimensional grammars and automata have been introduced in the literature [2, 11, 12, 20]. Two-dimensional matrix grammars [14], array grammars [14, 15], tiling systems [17, 18] are to mention a few. Automata working over a two-dimensional tape were firstly introduced by M. Blum and C. Hewitt [6, 7] in 1967. All these approaches were initially motivated by problems arising in the framework of pattern recognition and image processing. Two-dimensional patterns also appear in studies concerning cellular automata and some other models of parallel computing. The most common two-dimensional object is a picture which is a rectangular array of symbols taken from a finite alphabet. We restrict ourselves to the study of languages build from such objects. These languages are called picture (array) languages.

In the recent past, pure 2D context-free grammars [18] and their variants [20] were introduced which make use of only terminal symbols to generate rectangular array languages and allow rewriting of any (specific) row or column of the rectangular array. In this work, we draw some comparisons on the generative power of the pure 2D context-free grammars and its variants (l/u) pure 2D context-free grammar, (r/d) pure 2D context-free grammar with other interesting rectangular array grammars like Prusa Grammars, Parallel contextual array insertion and deletion grammars and Regional tile rewriting grammars which make use of terminals as well as non-terminal symbols in their derivation rules.

The thesis is organized as follows: In Chapter 1, some basic definitions of Prusa Grammar (PG) [8, 9, 10], Regional Tile rewriting grammar (RTG) [3], Parallel contextual array insertion and deletion grammars [12, 13, 14] and pure 2D context-free grammars (P2DCFG) [16, 17, 18, 19] and its variants, and few examples on corresponding languages are recalled. In Chapter 2, a literature review is undergone. In Chapter 3, the comparison of generative powers of P2DCFG with PG and PCAIDG are discussed. In Chapter 3, generative powers of variants of P2DCFG

(l/u) P2DCFG and (r/d) P2DCFG of P2DCFG are compared with PG and RTG. In Chapter 4, this thesis is concluded with some new problems.

PRELIMINARY

1.1 PURE 2D CONTEXT-FREE GRAMMAR

Based on the notion of pure context-free rules, a new two-dimensional grammar is introduced [16] for picture generation. The salient feature of this model is that the shearing effect in replacing a sub-array of a given rectangular array is taken care of by rewriting a row or a column of symbols in parallel by equal length strings and by using only terminal symbols as in a pure string grammar. We now recall the definition of Pure 2D Context-free grammar.

Definition 1: A pure 2D context-free grammar (P2DCFG) is a 4-tuple

$G = (\Sigma, P_c, P_r, \mathcal{M}_0)$, where

- Σ is a finite set of symbols.
- $P_c = \{t_{ci} \mid 1 \leq i \leq m\}$, $P_r = \{t_{rj} \mid 1 \leq j \leq n\}$.

Each t_{ci} , $(1 \leq i \leq m)$, called a column table, is a set of context-free rules of the form $a \rightarrow \alpha$, $a \in \Sigma$, $\alpha \in \Sigma^*$ such that for any two rules $a \rightarrow \alpha$, $b \rightarrow \beta$ in t_{ci} , we have $|\alpha| = |\beta|$,

where $|\alpha|$ denotes the length of α . Each t_{rj} , $(1 \leq j \leq n)$, called a row table, is a set of context-free rules of the form $c \rightarrow \gamma T$, $c \in \Sigma$ and $\gamma \in \Sigma^*$ such that for any two rules

$c \rightarrow \gamma T$, $d \rightarrow \delta T$ in t_{rj} , we have $|\gamma| = |\delta|$.

- $\mathcal{M}_0 \subseteq \Sigma^{**} - \{\lambda\}$ is a finite set of axiom arrays, where λ is empty word.

Derivations are defined as follows. For any two arrays M_1, M_2 , we write $M_1 \Rightarrow M_2$ if M_2 is obtained from M_1 by either rewriting a column of M_1 by rules of some column table t_{ci} in P_r or a row of M_1 by rules of some row table t_{rj} in P_r . $\Rightarrow^*$ is the reflexive transitive closure of $\Rightarrow$.

Example 1: Consider the pure 2D context-free grammar $G_1 = (\Sigma_1, P_{c1}, P_{r1}, \{M_{01}\})$ where

$$M_{01} = \begin{Bmatrix} x & \cdot & x \\ z & y & z \\ x & \cdot & x \end{Bmatrix}$$

$$t_{c1} = \{ \cdot \rightarrow \cdot \cdot, \quad y \rightarrow yy \}, \quad t_{r1} = \left\{ \begin{array}{cc} y \rightarrow \cdot & z \rightarrow x \\ & y, \quad z \\ & \cdot \quad x \end{array} \right\}$$

$$\begin{array}{ccccccc}
& & & & \text{X} & \cdot & \cdot & \text{X} & & \text{X} & \cdot & \cdot & \text{X} \\
& & & & & & & & & & & & & \\
& & & & \text{X} & \cdot & \cdot & \text{X} & & \text{X} & \cdot & \cdot & \text{X} \\
& & & & & & & & & & & & & \\
& & \text{X} & \cdot & \cdot & \text{X} & & \text{X} & \cdot & \cdot & \text{X} & & \text{X} & \cdot & \cdot & \text{X} \\
\mathbf{M}_{01} \xrightarrow{\quad} & \text{z} & \text{y} & \text{y} & \text{z} \Rightarrow & \text{z} & \text{y} & \text{y} & \text{z} \Rightarrow & \text{z} & \text{y} & \text{y} & \text{z} \Rightarrow & \text{z} & \text{y} & \text{y} & \text{z} & = & \mathbf{M}_1 \\
& \text{X} & \cdot & \cdot & \text{X} & & \text{X} & \cdot & \cdot & \text{X} & & \text{X} & \cdot & \cdot & \text{X} & & \text{X} & \cdot & \cdot & \text{X} \\
& & & & & & \text{X} & \cdot & \cdot & \text{X} & & \text{X} & \cdot & \cdot & \text{X} & & \text{X} & \cdot & \cdot & \text{X} \\
& & & & & & & & & & & \text{X} & \cdot & \cdot & \text{X} & & \text{X} & \cdot & \cdot & \text{X}
\end{array}$$

4

Example 2: Consider the pure 2D context-free grammar $G_2 = (\Sigma_2, P_{c2}, P_{r2}, \{M_{02}\})$, where

$$\Sigma_2 = \{x, y, z, \cdot\}, P_{c2} = \{t_{c2}\}, P_{r2} = \{t_{r2}\}$$

$$t_{c2} = \{y \rightarrow xyx, z \rightarrow \cdot z \cdot\} \quad t_{r2} = \left\{ \begin{array}{l} x \rightarrow x, y \rightarrow y \\ \cdot \quad \quad z \end{array} \right\} \quad M_{02} = \begin{pmatrix} x & y & x \\ \cdot & z & \cdot \end{pmatrix}$$

G_2 generates picture arrays M_2 of the form shown in Fig.2. Here, we note that the number of rows in the generated picture array need not have any proportion to the number of columns but will have an equal number of columns to the left and right of the middle column $(yz \dots z)^T$

$$\begin{array}{ccccccccc} x & x & x & y & x & x & x \\ \cdot & \cdot & \cdot & z & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & z & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & z & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & z & \cdot & \cdot & \cdot \end{array}$$

Fig.2. A picture array M_2 .

1.2 (l / u) PURE 2D CONTEXT FREE GRAMMAR

Definition 2: Let $G = (\Sigma, P_1, P_2, \mathcal{M}_0)$ be a P2DCFG with the components as in Definition 1. A (l/u) mode of the derivation of a picture array M_2 from M_1 in G , denoted by $\Rightarrow (l/u)$, is a derivation in G such that only the leftmost column or the uppermost row of M_1 is rewritten using respectively, the column rule tables or the row rule tables, to yield M_2 . The generated picture language is defined as in the case of a P2DCFG but with $\Rightarrow (l/u)$ derivations. The family of picture languages generated by P2DCFGs under $\Rightarrow (l/u)$ derivations is denoted by (l/u) P2DCFL. For convenience, we write (l/u) P2DCFG to refer to P2DCFG with $\Rightarrow (l/u)$ derivations.

Example 3

Consider an (l/u) P2DCFG , $G = (\Sigma, P_1, P_2, \{M_0\})$

$$M_0 = \begin{Bmatrix} b & a \\ a & b \end{Bmatrix}$$

$\Sigma = \{a, b\}, P_1 = \{c\}, P_2 = \{r\}$ with $c = \{a \rightarrow a b, b \rightarrow b a\}$,

$$r = \begin{Bmatrix} a \rightarrow a & , & b \rightarrow b \\ b & & a \end{Bmatrix}$$

The derivation of a picture starting from M_0 and using the tables c, c, r is shown below. We note that in this derivation (unlike in a derivation in a P2DCFG), the application of the column rule table c rewrites all symbols in the leftmost column in parallel and likewise, the application of the row rule table r rewrites all symbols in the uppermost row.

$$\begin{array}{ccccccc} M_0 = & b & a & \Rightarrow (l/u) & b & a & a & \Rightarrow (l/u) & b & a & a & a & \Rightarrow (l/u) & b & a & a & a \\ & a & b & & a & b & b & & a & b & b & b & & a & b & b & b \\ & & & & & & & & & & & & & b & b & b & b \end{array}$$

1.3 (r/d) PURE 2D CONTEXT FREE GRAMMAR

Definition 3: Let $G = (\Sigma, P_1, P_2, \Gamma)$ be a P2DCFG with the components as in Definition 1. An (r/d) mode of the derivation of a picture array M_2 from M_1 in G , denoted by $\Rightarrow(r/d)$, is a derivation in G such that **only the rightmost column or the lowermost row** of M_1 is rewritten using respectively, the column rule tables or the row rule tables, to yield M_2 . The generated picture language is defined as in the case of a P2DCFG but with $\Rightarrow(r/d)$ derivations. $\Gamma \subseteq \Sigma^{**} - \{\lambda\}$ is a finite set of axiom arrays.

The family of picture languages generated by P2DCFGs under $\Rightarrow(r/d)$ derivations is denoted by (r/d) P2DCFL. For convenience, we write (r/d) P2DCFG to refer to P2DCFG with $\Rightarrow(r/d)$ derivations.

EXAMPLE - 4

Consider an (r/d) P2DCFG: $G = (\Sigma, P_1, P_2, \{M_0\})$ where

$$M_0 = \begin{Bmatrix} b & a \\ a & b \end{Bmatrix}$$

$\Sigma = \{a, b\}$, $P_1 = \{c\}$, $P_2 = \{r\}$ with $c = \{a \rightarrow ba, b \rightarrow ab\}$

$r = \{a \rightarrow \begin{smallmatrix} b \\ a \end{smallmatrix}, b \rightarrow \begin{smallmatrix} a \\ b \end{smallmatrix}\}$. Starting from M_0 and using the tables in the order c, r, c, c the derivation of pictures by (r/d) P2DCFG G is shown below. We note that in this derivation (unlike in a derivation in a P2DCFG), the application of the column rule table c rewrites all symbols in the rightmost column in parallel and likewise, the application of the row rule table r rewrites all symbols in the lowermost row in parallel.

$$\begin{array}{ccccccc} & (r/d) & & (r/d) & & (r/d) & & (r/d) \\ M_0 = & b \ a & \Rightarrow & b \ b \ a & \Rightarrow & b \ b \ a & \Rightarrow & b \ b \ b \ a & \Rightarrow & b \ b \ b \ b \ a \\ & a \ b & & a \ a \ b & & b \ b \ a & & b \ b \ b \ a & & b \ b \ b \ b \ a \\ & & & & & a \ a \ b & & a \ a \ a \ b & & a \ a \ a \ a \ b \end{array}$$

1.4 PRUSA GRAMMAR

In the quest for generality, D. Pruša has recently defined a grammar model [10] that extends context-free Kolam rules [13], gaining some generative capacity. This grammar maintains the shape of the initial rectangular grid even if the array grows.

Definition 4: A 2D CF Pruša grammar (PG) is a tuple (Σ, N, S, R) , where Σ is the finite set of terminal symbols, disjoint from the set N of nonterminal symbols; $S \in N$ is the start symbol, and $R \subseteq N \times (N \cup \Sigma)^{++}$ is the set of rules

either be terminal rules which have a picture as the right part or variable rules in which, the right part is seen as a "grid", where nonterminal can be replaced by other pictures, but maintain its grid-like structure.

Example - 5: The Prusa grammar generating the language of pictures with one row and one column of b's in a background of a's where lines of b's will not be at the border is

$$G = (\Sigma, N, S, R) \quad \Sigma = \{a, b\} \quad N = \{H, A, V\}$$

$$S \rightarrow \begin{array}{ccc} A & V & A \\ H & b & H \\ A & V & A \end{array} \quad A \rightarrow AM \mid M \quad M \rightarrow \begin{array}{c} a \\ M \end{array} \mid a,$$

$$V \rightarrow \begin{array}{c} b \\ V \end{array} \mid b \quad H \rightarrow bH \mid b$$

$$S \rightarrow \begin{array}{ccc} A & V & A \\ H & b & H \\ A & V & A \end{array} \rightarrow \begin{array}{ccc} M & b & M \\ b & b & b \\ M & b & M \end{array} \rightarrow \begin{array}{ccc} a & b & a \\ b & b & b \\ a & b & a \end{array}$$

1.5 REGIONAL TILE REWRITING GRAMMAR (RTG)

Definition 5: A homogeneous partition is regional (HR) iff distinct (not necessarily adjacent) subdomains that have distinct labels. A picture p is regional if it admits an HR partition. A language is regional if all its pictures are so

A_1	A_1	B	A_2	A_2
A_1	A_1	B	A_2	A_2
D_1	D_1	B	D_2	D_2
A_3	A_3	C	A_4	A_4
A_3	A_3	C	A_4	A_4

Definition 6: A regional tile rewriting grammar (RTG) [3] is a tuple (Σ, N, S, R) , where Σ is the terminal alphabet, N is a set of nonterminal symbols, $S \in N$ is the starting symbol, R is a set of rules. Let $A \in N$. There are two kinds of rules:

Fixed-size: $A \rightarrow t$, where $t \in \Sigma$;

Variable size: $A \rightarrow \omega$, where ω is a set of tiles over $N \cup \{\#\}$ and

LOC (ω) is a regional language.

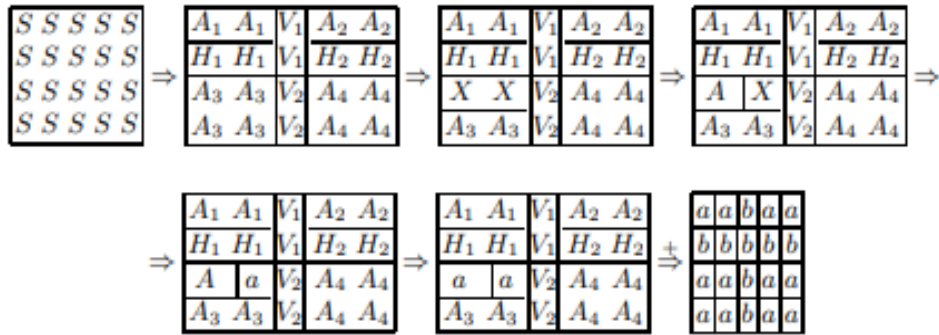
EXAMPLE - 6: The RTG grammar generating the language of pictures with one row and one column of b's, where lines of b's will not be at the border is

$$G = (\Sigma, N, S, R) \quad \Sigma = \{a, b\} \quad N = \{A_1, A_2, A_3, A_4, H_1, H_2, V_1, V_2, X\}$$

$$S \rightarrow \begin{bmatrix} \# & \# & \# & \# & \# & \# \\ \# & A_1 & A_1 & V_1 & A_2 & A_2 & \# \\ \# & A_1 & A_1 & V_1 & A_2 & A_2 & \# \\ \# & H_1 & H_1 & V_1 & H_2 & H_2 & \# \\ \# & A_3 & A_3 & V_2 & A_4 & A_4 & \# \\ \# & A_3 & A_3 & V_2 & A_4 & A_4 & \# \\ \# & \# & \# & \# & \# & \# & \# \end{bmatrix}; \quad A_i \rightarrow \begin{bmatrix} \# & \# & \# & \# \\ \# & X & X & \# \\ \# & A_i & A_i & \# \\ \# & A_i & A_i & \# \\ \# & \# & \# & \# \end{bmatrix} \mid \begin{bmatrix} \# & \# & \# & \# \\ \# & X & X & \# \\ \# & \# & \# & \# \end{bmatrix}, \text{ for } 1 \leq i \leq 4$$

$$X \rightarrow \begin{bmatrix} \# & \# & \# & \# & \# \\ \# & A & X & X & \# \\ \# & \# & \# & \# & \# \end{bmatrix} \mid a; \quad H_i \rightarrow \begin{bmatrix} \# & \# & \# & \# & \# \\ \# & B & H_i & H_i & \# \\ \# & \# & \# & \# & \# \end{bmatrix} \mid b, \text{ for } 1 \leq i \leq 2$$

$$A \rightarrow a; \quad B \rightarrow b; \quad V_i \rightarrow \begin{bmatrix} \# & \# & \# \\ \# & B & \# \\ \# & V_i & \# \\ \# & V_i & \# \\ \# & \# & \# \end{bmatrix} \mid b, \text{ for } 1 \leq i \leq 2.$$



1.6 PARALLEL CONTEXTUAL ARRAY INSERTION AND DELETION GRAMMAR

Let V be a finite alphabet, and V^* be the set of words over Σ including the empty word λ .

$V^+ = V^* - \{\lambda\}$. For $w \in V^*$ and $a \in V$, $|w|_a$ denotes the number of occurrences of a in w . An array consists of finitely many symbols from V that are arranged as rows and columns in some particular order and are written in the form,

Definition 7 . A parallel contextual array insertion-deletion grammar is defined as $G = (V, T, M, C, R, \phi_c^i, \phi_r^i, \phi_c^d, \phi_r^d)$, where V is an alphabet, $T \subseteq V$ is a terminal alphabet, M is a finite subset of V^{**} called the base of G , C is a finite subset of V^{**} called column array contexts, R is a finite subset of V^{**} called row array contexts, $\phi_c^i : V^{**} \times V^{**} \rightarrow 2^C$, $\phi_r^i : V^{**} \times V^{**} \rightarrow 2^R$, $\phi_c^d : V^{**} \times V^{**} \rightarrow 2^C$, $\phi_r^d : V^{**} \times V^{**} \rightarrow 2^R$, are the choice mappings which perform the parallel column contextual insertion, row contextual insertion, column contextual deletion, and row contextual deletion operations, respectively.

Example 7 : Consider a grammar

$G = \langle V, T, M, C, R, \phi_c^i, \phi_r^i, \phi_c^d, \phi_r^d \rangle$ where

$$V = \{x, y, z, \bullet, u, w\} \quad T = \{x, y, z, \bullet\} \quad C = \left\{ \begin{array}{ccccc} x & u & w & x & w & u \\ \bullet & u & w & \bullet & w & u \end{array} \right\}$$

$$R = \left\{ \begin{array}{ccccccccc} \bullet & \bullet & \bullet & z & z & \bullet & w & w & u & u \\ u & u & u & u & u & u & \bullet & u & & \end{array} \right\} \quad \text{respectively}$$

$$\left\{ y \rightarrow xyx, z \rightarrow \bullet z \right\} \quad \left\{ x \rightarrow \begin{array}{c} x \\ \bullet \end{array}, y \rightarrow \begin{array}{c} y \\ z \end{array} \right\}$$

$$M = \left\{ \begin{array}{ccc} x & y & x \\ \bullet & z & \bullet \end{array} \right\}$$

Insertion & Deletion Rules

$$\text{Rule 1 : } \phi_c^i \left\{ \begin{array}{cc} x & y & x \\ \bullet & z & \bullet \end{array} \right\} = \begin{array}{cc} x & u \\ \bullet & u \end{array}$$

$$\text{Rule 2 : } \phi_c^i \left\{ \begin{array}{ccc} u & y & x \\ u & z & \bullet \end{array} \right\} = \begin{array}{cc} w & x \\ w & \bullet \end{array}$$

$$\text{Rule 3: } \phi_c^d \left\{ \begin{array}{cc} x & y & w \\ \bullet & z & w \end{array} \right\} = \begin{array}{cc} u & \\ & u \end{array}$$

$$\text{Rule 4: } \phi_c^d \left\{ \begin{array}{ccc} x & y & x \\ \bullet & z & \bullet \end{array} \right\} = \begin{array}{cc} w & \\ & w \end{array}$$

Derivation of a T shaped picture by the PCAIDG G is shown below

x y x
 . z .



x x u y z
 . . u z .



x y w x x
 . z w . .



x x u y w x x
 . . u z w . .



x x y x x
 . . z . .



x x x u y x x
 . . . u z . .



x x x y w x x x
 . . . z w . . .



x x x u y w x x x
 . . . u z w . . .



x x x y x x x
 . . . z . . .

CHAPTER - 2

LITERATURE SURVEY

The study of two-dimensional picture languages has picked pace from 1970's for solving problems arising in the framework of pattern recognition and image processing by extending the theory of one-dimensional string languages to two-dimensional picture languages.

Motivated by certain floor designs called patterns, Siromoney et al. [1972] [14] introduced Siromoney Matrix Grammars (SMG), a simple and elegant grammar model for generating rectangular arrays. This is one of the earliest picture models and has been widely investigated for its theoretical properties and applications. Siromoney matrix grammars are sequential-parallel, in the sense that first a horizontal string of nonterminals is derived sequentially, using the horizontal productions; and then the vertical derivations proceed in parallel, applying a set of vertical productions. Recently, more powerful context-free picture grammars called, Prusa grammars [8, 9, 10] [Prusa 2004] and Tile Rewriting Grammars [3] [Crespi 2003] are introduced; In the former case the right part of the rules is seen as a grid where non-terminals can be replaced by other pictures, but maintaining its grid-like structure, and the latter case combines [1979] isometric rewriting rules with the Tiling system of Giammerresi [4,5] and Restivo [1997].

Based on pure context-free string grammars which make use of only terminal symbols, a two-dimensional grammar model, called pure 2D context-free picture grammar (P2DCFG) [17, 18, 19] is introduced in [Subramanian 2008a, 2009]. Unlike the models in SMG, rewriting any column or any row of the rectangular array is allowed in this model. The generative power of P2DCFG was investigated along with certain closure properties. Subsequently, a variant $(l/u)P2DCFG$ of P2DCFG [11, 12] with the rewriting of only the leftmost column or the uppermost row of a picture array was considered, and the properties of the resulting family $L((l/u)P2DCFG)$ of picture languages were obtained. Another variant of $(r/d)P2DCFG$ [11, 12] with the rewriting of only the rightmost column or the lowermost row of a picture array was considered and properties of the resulting family $L((r/d)P2DCFG)$ of picture languages were obtained. Unlike the case of string context-free grammars, the picture language families of P2DCFG and the two variants $(l/u)P2DCFG$ and $(r/d)P2DCFG$ are mutually incomparable, although they are not disjoint [11].

CHAPTER - 3

COMPARISON OF PURE 2D CONTEXT-FREE GRAMMARS WITH PRUSA AND PARALLEL CONTEXTUAL ARRAY INSERTION AND DELETION GRAMMARS

In the following sequel, $\mathcal{L}[X]$ denotes the family of all languages generated by the grammar X .

Theorem 3.1: $\mathcal{L}[P2DCFG] \cap \mathcal{L}[PG] \neq \Phi$

Proof: Consider the language

$$L_1 = \left\{ \begin{array}{ccc} x & y & x \\ \cdot & z & \cdot \end{array} , \quad \begin{array}{ccccc} x & x & y & x & x \\ \cdot & \cdot & z & \cdot & \cdot \end{array} , \quad \begin{array}{ccc} x & y & x \\ \cdot & z & \cdot \end{array} , \quad \dots \right\}$$

of all T shaped pictures where the horizontal limb of T is made of x's with middle y and vertical limb of T is consisting of z's below y and other positions are occupied by

•

The P2DCFG for generating L_1 is given as follows:

$$G_1 = \langle \Sigma_1, P_{c1}, P_{r1}, \{M_0\} \rangle$$

$$\Sigma = \{x, y, z, \cdot\} \quad P_{c1} = \{t_{c1}\} \quad P_{r1} = \{t_{r1}\}$$

$$t_{c1} = \{y \rightarrow xyx, z \rightarrow \cdot z \cdot\} \quad t_{r1} = \left\{ \begin{array}{cc} x \rightarrow x & y \rightarrow y \\ \cdot & z \end{array} \right\} \quad M_{01} = \left\{ \begin{array}{ccc} x & y & x \\ \cdot & z & \cdot \end{array} \right\}$$

G_1 generating a picture array M_1 from M_0 is shown below

$$M_0 = \begin{array}{ccc} x & y & x \\ \cdot & z & \cdot \end{array} \rightarrow \begin{array}{ccccc} x & x & y & x & x \\ \cdot & \cdot & z & \cdot & \cdot \end{array} \rightarrow \begin{array}{ccccc} x & x & y & x & x \\ \cdot & \cdot & z & \cdot & \cdot \\ \cdot & \cdot & z & \cdot & \cdot \end{array} \rightarrow \begin{array}{ccccc} x & x & y & x & x \\ \cdot & \cdot & z & \cdot & \cdot \\ \cdot & \cdot & z & \cdot & \cdot \\ \cdot & \cdot & z & \cdot & \cdot \end{array} = M_1$$

The Prusa Grammar (PG) generating L_1 is given as:

$$G_2 = \langle \Sigma, N, S, R \rangle$$

$$\Sigma = \{x, y, z, \cdot\}, \quad N = \{H, B, V, A\} \quad S = \left\{ \begin{array}{ccc} x & H & x \\ A & V & A \end{array} \right\}$$

$$R = A \rightarrow \begin{array}{c} A \\ \cdot \end{array} \mid \cdot$$

$$V \rightarrow \begin{array}{c} V \\ B \end{array} \mid z \mid AVA$$

$$B \rightarrow MBM \mid z$$

$$H \rightarrow XHX \mid y$$

$$M \rightarrow \cdot$$

$$X \rightarrow x$$

Derivation of the picture

x	x	x	y	x	x	x
.	.	.	z	.	.	.
.	.	.	z	.	.	.

$$S \rightarrow \begin{array}{ccc} x & H & x \\ A & V & A \end{array} \rightarrow \begin{array}{ccccc} b & b & H & b & b \\ A & A & V & A & A \end{array} \rightarrow \begin{array}{ccccc} x & X & H & X & x \\ A & A & V & A & A \end{array} \rightarrow \begin{array}{ccccc} x & x & H & x & x \\ A & A & V & A & A \\ . & . & B & . & . \end{array}$$

$\downarrow$		$\leftarrow$
$\begin{array}{ccccccc} x & x & x & y & x & x & x \\ . & . & . & z & . & . & . \\ . & . & . & z & . & . & . \end{array}$		$\begin{array}{ccccccc} x & x & x & y & x & x & x \\ . & . & . & z & . & . & . \\ . & . & . & z & . & . & . \end{array}$

So, $\mathcal{L} [P2DCFG] \cap \mathcal{L} [PG]$ contains the language of T-shaped pictures. Hence
 $\mathcal{L} [P2DCFG] \cap \mathcal{L} [PG] \neq \Phi$

Theorem 3.2: $\mathcal{L} [PG] \not\subseteq \mathcal{L} [P2DCFG]$

Proof: Consider the language

$$L_2 = \left\{ \begin{array}{ccc} b & b & b \\ a & b & a \end{array} , \quad \begin{array}{ccccc} b & b & b & b & b \\ a & a & b & a & a \end{array} , \quad \begin{array}{ccccc} b & b & b & b & b \\ a & a & b & a & a \\ a & a & b & a & a \end{array} \dots \right\}$$

of all T-shaped pictures with limbs consisting of b's and other positions are occupied by a.

The Prusa Grammar (PG) generating L_2 is given as:

$$G_3 = \langle \Sigma, N, S, R \rangle$$

$$S \rightarrow \begin{array}{ccc} b & H & b \\ A & V & A \end{array} \quad \Sigma = \{a, b\}, \quad N = \{H, V, A\}$$

$$A \rightarrow A \quad V \rightarrow A$$

$$R = \begin{array}{c} A \rightarrow A \quad a \\ a \quad | \end{array}$$

$$V \rightarrow \begin{array}{c} V \quad | \quad AVA \\ B \quad | \end{array} b$$

$$B \rightarrow bBb \quad | \quad Bb \quad | \quad b$$

$$H \rightarrow bHb \quad | \quad b$$

Derivation of the picture $\begin{array}{ccccc} b & b & b & b & b \\ a & a & b & a & a \\ a & a & b & a & a \\ a & a & b & a & a \\ a & a & b & a & a \end{array}$ is given below

$$\begin{array}{ccccc} b & b & b & b & b \\ a & a & b & a & a \\ a & a & b & a & a \\ a & a & b & a & a \\ a & a & b & a & a \end{array}$$

$$\begin{array}{ccccccc}
S \rightarrow & b & H & b & \rightarrow & b & b & H & b & b & \rightarrow & b & b & b & b & b & \rightarrow & b & b & b & b & b \\
& A & V & A & & A & A & V & A & A & & A & A & V & A & A & & A & A & V & A & A \\
& & & & & & & & & & & a & a & B & a & a & & a & a & B & a & a \\
& & & & & & & & & & & & & & & & a & a & b & a & a
\end{array}$$

$$\begin{array}{ccccc}
b & b & b & b & b \\
a & a & b & a & a \\
a & a & b & a & a \\
a & a & b & a & a \\
a & a & b & a & a
\end{array}$$


$$\begin{array}{ccccc}
b & b & b & b & b \\
A & A & V & A & A \\
a & a & B & a & a \\
a & a & b & a & a \\
a & a & b & a & a
\end{array}$$

But P2DCFG for generating L_2 is not possible:

Suppose if we try to construct a P2DCFG grammar, $G = \langle \Sigma_1, P_{c1}, P_{r1}, M_{01} \rangle$

$$M_{01} = \left\{ \begin{array}{ccc} b & b & b \\ a & b & a \end{array} \right\}$$

$$\Sigma = \{a, b\} \quad P_{c1} = \{t_{c1}\} \quad P_{r1} = \{t_{r1}\}$$

$$t_{c1} = \{b \rightarrow b b b\} \quad t_{r1} = \left\{ \begin{array}{cc} b \rightarrow b, & b \rightarrow b \\ & a \qquad \qquad b \end{array} \right\}$$

Application of the above rules will not yield L_2 . A picture not in L_2 can also be generated is evident from the following derivation, by applying t_{c1} on M_{01}

$$\begin{array}{ccccc}
b & b & b & \rightarrow & b & b & b & b & b \\
a & b & a & & a & b & b & b & a
\end{array}$$

So, $\mathcal{L} [PG] \not\subseteq \mathcal{L} [P2DCFG]$.

Theorem 3.3: $\mathcal{L} [P2DCFG] \cap \mathcal{L} [PCAIDG] \neq \Phi$

Proof: Consider the language

$$L_1 = \left\{ \begin{array}{ccc} x & y & x \\ \cdot & z & \cdot \end{array} , \begin{array}{ccccc} x & x & y & x & x \\ \cdot & \cdot & z & \cdot & \cdot \end{array} , \begin{array}{ccc} x & y & x \\ \cdot & z & \cdot \\ \cdot & z & \cdot \end{array} , \dots \right\}$$

of all T shaped pictures where the horizontal limb of T is made of x's with middle y and vertical limb of T is consisting of z's below y and other positions occupied by $\cdot$.

The P2DCFG for generating L_1 is given in the proof of Theorem 3.1:

The PCAIDG for generating L_1 is given as follows:

$$G_4 = \langle V, T, M, C, R, \phi_c^i, \phi_r^i, \phi_c^d, \phi_r^d \rangle \quad \text{where}$$

$$V = \{x, y, z, \cdot, u, w\} \quad T = \{x, y, z, \cdot\} \quad C = \left\{ \begin{array}{cccccc} x & u & w & x & \cdot & u & u \\ \cdot & u & w & \cdot & \cdot & u & u \end{array} \right\}$$

$$R = \left\{ \begin{array}{cccccc} \cdot & \cdot & \cdot & z & z & \cdot & u & u \\ u & u & u & u & u & u & u & u \end{array} \right\} \quad \text{respectively.}$$

$$\left\{ y \rightarrow xyx, z \rightarrow \cdot z \cdot \right\} \quad \left\{ x \rightarrow \begin{array}{c} x \\ \cdot \end{array} , y \rightarrow \begin{array}{c} y \\ z \end{array} \right\}$$

$$M = \left\{ \begin{array}{ccc} x & y & x \\ \cdot & z & \cdot \end{array} \right\}$$

Insertion & Deletion Rules

$$\text{Rule 1 : } \phi_c^i \left\{ \begin{array}{ccc} x & , & y & x \\ \cdot & & z & \cdot \end{array} \right\} = \begin{array}{cc} x & u \\ \cdot & u \end{array}$$

$$\text{Rule 2 : } \phi_c^i \left\{ \begin{array}{ccc} u & y & , & x \\ u & z & & \cdot \end{array} \right\} = \begin{array}{cc} w & x \\ w & \cdot \end{array}$$

$$\text{Rule 3: } \phi_c^d \left\{ \begin{array}{cc} x & , & y w \\ . & & z w \end{array} \right\} = \begin{array}{c} u \\ u \end{array}$$

$$\text{Rule 4: } \phi_c^d \left\{ \begin{array}{ccc} x & y & , & x \\ . & z & & . \end{array} \right\} = \begin{array}{c} w \\ w \end{array}$$

$$\text{Rule 5: } \phi_r^i \left\{ \begin{array}{ccc} x & x & , & . & . \\ & & & u & u \end{array} \right\} = \begin{array}{c} . & . \\ u & u \end{array}$$

$$\text{Rule 6: } \phi_r^i \left\{ \begin{array}{ccc} x & y & , & . & z \\ & & & u & u \end{array} \right\} = \begin{array}{c} . & z \\ u & u \end{array}$$

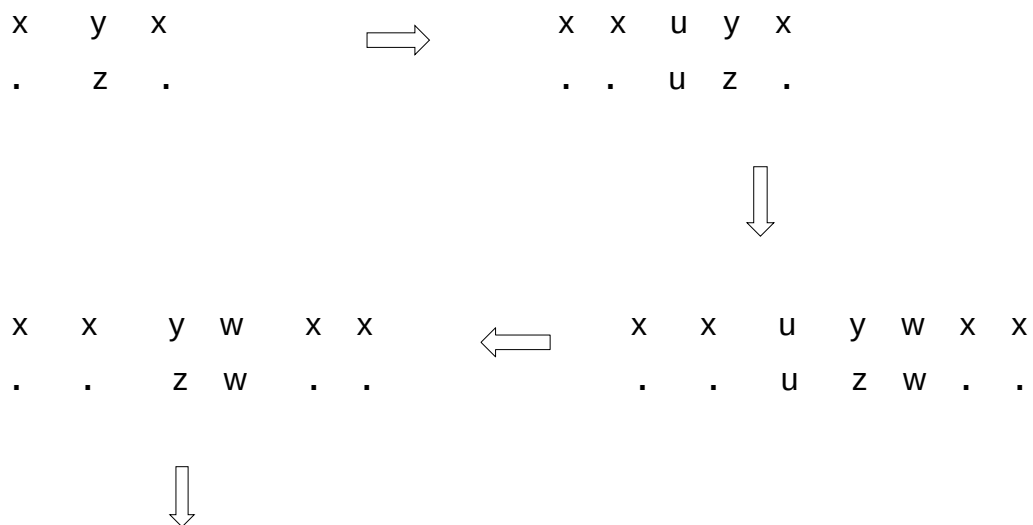
$$\text{Rule 7: } \phi_r^i \left\{ \begin{array}{ccc} y & x & , & z & . \\ & & & u & u \end{array} \right\} = \begin{array}{c} z & . \\ u & u \end{array}$$

$$\text{Rule 8: } \phi_r^d \left\{ \begin{array}{ccc} . & . & , & . & . \\ & & & u & u \end{array} \right\} = \begin{array}{c} u & u \\ u & u \end{array}$$

$$\text{Rule 9: } \phi_r^d \left\{ \begin{array}{ccc} . & z & , & . & z \\ & & & u & u \end{array} \right\} = \begin{array}{c} u & u \\ u & u \end{array}$$

$$\text{Rule 10: } \phi_r^d \left\{ \begin{array}{ccc} z & . & , & z & . \\ & & & u & u \end{array} \right\} = \begin{array}{c} u & u \\ u & u \end{array}$$

Derivation a typical picture in L_1 by the rules of G_4 is shown below



x	x	y	x	x		x	x	x	u	y	x	x
.	.	z	.	.		.	.	.	u	z	.	.



x	x	x	y	w	x	x	x		x	x	x	u	y	w	x	x	x
.	.	.	z	w	.	.	.		.	.	.	u	z	w	.	.	.



x	x	x	y	x	x	x
.	.	.	z	.	.	.

So, $\mathcal{L}[\text{P2DCFG}] \cap \mathcal{L}[\text{PCAIDG}]$ contains the language L_1 .

Hence $\mathcal{L}[\text{P2DCFG}] \cap \mathcal{L}[\text{PCAIDG}] \neq \Phi$

CHAPTER- 4

COMPARISON OF VARIANTS OF PURE 2D CONTEXT-FREE GRAMMARS WITH PRUSA GRAMMARS AND REGIONAL TILE REWRITING GRAMMARS

Theorem 4.1: $\mathcal{L} [(l/u) \text{ P2DCFG}] \cap \mathcal{L} [\text{PG}] \neq \Phi$

Proof: Consider a language

$$L_3 = \left\{ \begin{array}{ccc} b & a & a, & b & a & a, & b & a & a & a, & \dots\dots\dots \\ a & b & b & a & b & b & a & b & b & b \\ a & b & b & a & b & b & a & b & b & b \\ & & & a & b & b & a & b & b & b \end{array} \right\}$$

Consider a (l/u) P2DCFG

$G_5 = (\Sigma, P_1, P_2, \{M_0\})$ where

$\Sigma = \{a, b\}, P_1 = \{c\}, P_2 = \{r\}$

$$M_0 = \left\{ \begin{array}{cc} b & a \\ a & b \end{array} \right\}$$

$$c = \{a \rightarrow a b, b \rightarrow b a\}, \quad r = \left\{ \begin{array}{cc} a \rightarrow a, & b \rightarrow b \\ & b \quad \quad a \end{array} \right\}$$

G_5 generates picture array L_3 as shown in the figure:

$$M_0 = \begin{array}{cccc} b & a \rightarrow & b & a & a \rightarrow & b & a & a \rightarrow & b & a & a \\ & a & b & & a & b & b & & a & b & b \\ & & & & a & b & b & & a & b & b \\ & & & & & & & & a & b & b \end{array}$$

The Prusa Grammar (PG) generating L_3 is given as:

$$G_6 = \langle \Sigma, N, S, R \rangle$$

$$\Sigma = \{a, b\}, \quad N = \{A, B, C, V\},$$

$$S \rightarrow SA \mid S \mid S A \mid b a \\ \quad \quad \quad \quad \quad C \mid C B \mid a b$$

$$R = A \rightarrow a \mid A U \mid a \\ \quad \quad \quad B$$

$$B \rightarrow b \mid b B \mid b \\ \quad \quad \quad B$$

$$C \rightarrow a B \mid C \mid a \\ \quad \quad \quad V$$

$$V \rightarrow C$$

$$U \rightarrow A$$

A picture array in L_3 generated by G_6 as shown below:

$$\begin{array}{ccccccc} S \rightarrow S & A \rightarrow S & A \rightarrow S & A \rightarrow b & a & a \rightarrow b & a & a \\ & C & B & C & B & C & b & a & b & B & a & b & b \\ & & & V & B & C & B & c & b & B & a & b & b \\ & & & & & & & a & b & b & a & b & b \end{array}$$

$$\text{So, } L_3 \in \mathcal{L}[(l/u) \text{ P2DCFG}] \cap \mathcal{L}[\text{PG}]$$

Hence,

$$\mathcal{L}[(l/u) \text{ P2DCFG}] \cap \mathcal{L}[\text{PG}] \neq \Phi$$

Theorem 4.2: $\mathcal{L} [\text{PG}] \not\subseteq \mathcal{L} [(l/u) \text{ P2DCFG}]$

Proof: Consider the language

$$L_2 = \left\{ \begin{array}{ccc} b & b & b & b & b, & b & b & b, & \dots\dots\dots \\ a & a & b & a & a & a & b & a \\ & & & & & a & b & a \end{array} \right\}$$

of all T-shaped pictures with limbs consisting of b's, other positions are occupied by a.

The Prusa Grammar (PG) generating the language L_2 is given as given Theorem 3.2 in Chapter 3

But this kind of picture pattern is not generated by (l/u) P2DCFG since in this case grammar rules can only be applied in the leftmost and uppermost positions. So we can say that the family of languages generated by PG is not contained in the family of languages generated by (l/u) P2DCFG

Theorem 4.3: $\mathcal{L} [(r/d) \text{ P2DCFG}] \cap \mathcal{L} [\text{PG}] \neq \Phi$

Proof: Consider a language

$$L_4 = \left\{ \begin{array}{ccc} b & b & a, & b & b & a, & b & b & b & a, & \dots\dots\dots \\ a & a & b & b & b & a & b & b & b & a \\ & & & b & b & a & b & b & b & a \\ & & & a & a & b & a & a & a & b \end{array} \right\}$$

Consider an (r/d) P2DCFG

$G_7 = (\Sigma, P_1, P_2, \{M_0\})$ where

$\Sigma = \{a, b\}, P_1 = \{c\}, P_2 = \{r\}$

$$M_0 = \left\{ \begin{array}{cc} b & a \\ a & b \end{array} \right\}$$

$$c = \{a \rightarrow b a, b \rightarrow a b\}, r = \left\{ \begin{array}{cc} a \rightarrow b, & b \rightarrow a \\ a & b \end{array} \right\}$$

A typical picture generated by G_7 is shown below:

$$M_0 = \begin{array}{ccccc} b & a & \rightarrow & b & b & a \\ & a & b & a & a & b \end{array}$$

The Prusa Grammar (PG) generating L_4 is given as:

$$G_8 = \langle \Sigma, N, S, R \rangle$$

$$\Sigma = \{a, b\}, \quad N = \{A, B, C, V\},$$

$$S \rightarrow \begin{array}{c|c|c|c} A & S & C & \\ \hline & & S & \end{array} \begin{array}{c|c|c|c} A & S & b & a \\ \hline & & a & b \end{array}$$

$$R = \begin{array}{c|c|c} A \rightarrow B & U A & a \\ \hline & a & \end{array}$$

$$B \rightarrow \begin{array}{c|c|c} B & B b & b \\ \hline a & & \end{array}$$

$$C \rightarrow \begin{array}{c|c|c} B a & V & a \\ \hline & C & \end{array}$$

$$V \rightarrow C$$

$$U \rightarrow A$$

The derivation of a picture array in L_4 generated by G_7 is as shown below:

$$\begin{array}{ccccccc}
 S \rightarrow A & S \rightarrow B & b & a & \rightarrow & b & b & a \\
 & a & a & b & & a & a & b
 \end{array}$$

So, $L_4 \in \mathcal{L}[(r/d) \text{ P2DCFG}] \cap \mathcal{L}[\text{PG}]$

Hence,

$$\mathcal{L}[(r/d) \text{ P2DCFG}] \cap \mathcal{L}[\text{PG}] \neq \Phi$$

Theorem 4.4: $\mathcal{L}[\text{PG}] \not\subseteq \mathcal{L}[(l/u) \text{ P2DCFG}]$

Proof: Consider the language

$$L_2 = \left\{ \begin{array}{ccccccc}
 b & b & b & b & b, & b & b & b, & \dots\dots\dots \\
 a & a & b & a & a & a & b & a \\
 & & & & & a & b & a
 \end{array} \right\}$$

of all T-shaped pictures with limbs consisting of b's, other positions are occupied by a.

The Prusa Grammar (PG) generating the language L_2 is given in the proof of Theorem 3.2 in Chapter 3

But this kind of picture pattern is not generated by (r/d) P2DCFG since in this case grammar rules can only be applied in the rightmost and down most positions. So we can say that the family of languages generated by PG is not contained the family of languages generated by (r/d) P2DCFG.

COMPARISON OF (l/u) and (r/d) P2DCFG WITH RTG

Theorem 4.5: $\mathcal{L}[(l/u) \text{ P2DCFG}] \cap \mathcal{L}[\text{RTG}] \neq \emptyset$

Theorem 4.6: $\mathcal{L}[\text{RTG}] \not\subseteq \mathcal{L}[(l/u) \text{ P2DCFG}]$

Theorem 4.7: $\mathcal{L}[(r/d) \text{ P2DCFG}] \cap \mathcal{L}[\text{RTG}] \neq \emptyset$

Theorem 4.8: $\mathcal{L}[\text{RTG}] \not\subseteq \mathcal{L}[(r/d) \text{ P2DCFG}]$

Theorem 4.5, 4.6, 4.7 and 4.8 are followed from the result $\mathcal{L}[\text{PG}] \subset \mathcal{L}[\text{RTG}]$ [3]

CONCLUSION

We have compared the generative powers of Pure 2D context-free grammars (P2DCFG) with Prusa grammar(PG) and Parallel contextual array insertion and deletion grammars (PCAIDG) and proved that family of languages $\mathcal{L}(\text{PG})$ and $\mathcal{L}(\text{PCAIDG})$ are not properly contained in the family of languages $\mathcal{L}(\text{P2DCFG})$ but have a non-empty intersection with it. We also compared PG and Regional Tile rewriting grammars (RTG) with two variants of P2DCFG: $((l/u)\text{P2DCFG})$ and $((r/d)\text{P2DCFG})$ and proved that $\mathcal{L}(\text{PG})$, $\mathcal{L}(\text{RTG})$ and $\mathcal{L}((l/u)\text{P2DCFG})$ and $\mathcal{L}((r/d)\text{P2DCFG})$ are not disjoint and neither $\mathcal{L}(\text{PG})$ nor $\mathcal{L}(\text{RTG})$ is properly contained in $\mathcal{L}((l/u)\text{P2DCFG})$ as well as $\mathcal{L}((r/d)\text{P2DCFG})$. It is worth investigating whether $\mathcal{L}((l/u)\text{P2DCFG})$ and $\mathcal{L}((r/d)\text{P2DCFG})$ are properly included in $\mathcal{L}(\text{PG})$ and $\mathcal{L}(\text{RTG})$ or not. The connections of these grammars with pattern generation and image analysis should be deeply explored. These are open problems for future work.

REFERENCES

- [1] Bersani, M.M., Frigeri, A., Cherubini, A.: On some classes of 2D languages and their relations. In: Aggarwal, J.K., Barneva, R.P., Brimkov, V.E., Koroutchev, K.N., Korutcheva, E.R. eds. IW CIA 2011. LNCS, vol. 6636, Springer, Heidelberg 2011 pp. 222---234.
- [2] Bordhin H, Fernau H, Holzer M, Accepting pure grammars and systems, Publicationes Mathematicae Debrecen 60 (Supplement) (2002) 483–510
- [3] Cherubini A, Crespi Reghizzi S, Pradella M, Regional languages, and tiling: A unifying approach to picture grammars, in Mathematical Foundations of Computer Science (MFCS 2008), Lecture Notes in Computer Science, vol. 5162, 2008, Springer, pp. 253–264
- [4] Giammarresi D, A. Restivo, Two-dimensional Languages In: G. Rozenberg, A. Saloma two dimensional Languages, in: G. Rozenberg, A. Salomaa (Eds.), Handbook of Formal Languages, vol. 3, Springer Verlag, 1997, pp. 215–267
- [5] Giammarresi D, Restivo A, Recognizable picture languages, in: Nivat M, Saoudi A, Wang P. S. P. (Eds.), Parallel Image Processing, Int. J. Pattern Recognit. Artif. Intell. (1992) 31–46. (special issue).
- [6] Jean – Camille Birget. Intersection and Union of regular language and state complexity. Information Processing Letters, 43(4), 1992 : 185-190.
- [7] Manuel Blum and Carl Hewitt. Automata on a 2-dimensional tape. In Miller R.E, editor, Proceedings of the 8th Annual Symposium on Switching and Automata Theory (SWAT 1967), 1967, pages. 155-160.

- [8] Průša D, Mráz F: Parallel Turing machines on a two-dimensional tape, In proceedings of the Czech Pattern Recognition Workshop, CVUT, Peršlák, ˇ 2000.
- [9] Průša D: Recognition of palindromes, In proceedings of WDS 2000, Charles University, Prague, 2000.
- [10] Průša D: Two-dimensional Context-free Grammars, In proceedings of ITAT 2001, Zuberec, 2001, pp. 27–40.
- [11] Rosenfeld A, Picture Languages — Formal Models for Picture Recognition, Academic Press, 1979, New York
- [12] Rosenfeld A, Siromoney R, Picture languages — A survey, Lang. Des. 1 (1993) 229–245.
- [13] Siromoney G, Siromoney R, Krithivasan K, Array grammars kolam, Compt. Graph. Image Process. 3, 1974 , 63–82.
- [14] Siromoney G ,Siromoney R, Krithivasan K, Abstract families of matrices and picture languages, Compt.Graph.ImageProcess.1(1972)234–307.
- [15] Siromoney G .Siromoney R, Krithivasan K, Picture languages with array rewriting rules,Inf.Control 22, 1973, 447–470.
- [16] Subramanian K.G., Geethalakshmi M, Gnanamalar David N, and Atulya K. Nagar: Picture Array Generation Using Pure 2D Context-Free Grammar Rules, R.P. Barneva et al. (Eds.): IWCI 2015, LNCS 9448, Springer, 2015, pp. 187–201
- [17] Subramaniana K.G, Rosihan M.Ali, Geethalakshmi M, Atulya K.Nagar, Pure 2D picture grammars and languages. Discrete Applied Mathematics, Elsevier, 2009, Vol 157, pp. 3401 – 3411

- [18] Subramanian, K.G., Ali, R.M., Geethalakshmi, M., Nagar, A.K.: Pure 2D picture grammars and languages. *Discrete Appl. Math.* 157,(2009) 3401–3411
- [19] Subramanian, K.G., Van, D.L., Helen Chandra, P., Quyen, N.D.: Array grammars with contextual operations. *Fundam. Inform.* 83, (2008) 411–428
- [20] Wang P.S.P. (Ed.), *Array Grammars, Patterns and Recognizers: Series in Computer Science*, vol.18, 1989, World Scientific
- [21] Zbynek Krivka, Carlos Mart'ın-Vide2, Alexander Meduna, and Subramanian K.G. : A Variant of Pure Two-Dimensional Context-Free Grammars Generating Picture Languages, R.P. Barneva et al. (Eds.): *IWCIA 2015, LNCS 8466*, Springer, 2014, pp. 123–133