A class of (R:R) array languages

D.G.Thomas, a S.Jayasankar b and T.Kamaraj c

a Department of Mathematics, Madras Christian College, Chennai – 600059, Tamil Nadu, India.
b Department of Mathematics, RKM Vivekananda College, Chennai- 620004, Tamil Nadu, India
c Department of Mathematics, Satyabama University, Chennai – 600119, Tamil Nadu, India.

Abstract

In this work we consider a restricted version of (R:R) Array Grammars [Siromoney 1973] with only right(left)-linear non-terminal rules involving column or row catenation operations. We establish some interesting results on closure properties of the family of languages generated by these grammars under standard array operations. We also compare generating power of these classes of languages with (R:R) Array Languages and Regular Matrix Languages [Siromoney 1972].

Keywords: Array languages, Array grammars.

2010 MSC: 05C20, 05C69.

I. Introduction

Formal phrase structure grammars in the Chomskian hierarchy together with a finite number of right linear grammars have provided a basis for generating matrices. The models are found useful to generate a variety of interesting classes of pictures. Simple transformations of a picture such as reflection, half-turn, and conjugation are easily obtained from the grammar itself. Chomskian hierarchy of phrase structure languages (PSL) induces a natural hierarchy among picture classes in the sense that there are distinct picture classes generated by one type but not generated by a grammar of a lower type. These matrices which generate matrices are generalisations of equal matrix grammars studied earlier [Siromoney, 1969]. Motivated by the work of R.Siromoney et al, in this paper a study about a sub-class of (R:R)AL[2] languages namely $(R_s:R)$ has been made.

II. Preliminaries

In this section we first review some of the definitions and results in Siromoney, Siromoney, and Krithivasan [1].

Notation: Let I be an alphabet-a finite nonempty set of symbols. A matrix over I is an m x n rectangular array of symbols from I (m, n ≥ 1). The set of all matrices over I (including $\Lambda$) is denoted by $I^{**}$ and $I^{++} = I^{**} - \{\Lambda\}$.

For strings $x$ and $y$, $x = a_1 ... a_n$ and $y = b_1 ... b_m$, the concatenation (product) of $x$ and $y$ is $x.y = x = a_1 ... a_n$ $y = b_1 ... b_m$. For matrices, we define two types of concatenation, viz., row and column catenation (row and column product).

Definition 2.1 If

$$X = \begin{bmatrix} a_{11} & ... & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & ... & a_{mn} \end{bmatrix}, \quad Y = \begin{bmatrix} b_{11} & ... & b_{1n'} \\ \vdots & & \vdots \\ b_{m'1} & ... & b_{m'n'} \end{bmatrix}$$

$X \cdot Y = \begin{bmatrix} a_{11} \cdot b_{11} + \ldots + a_{1n} \cdot b_{1n} \\ \vdots \\ a_{m1} \cdot b_{11} + \ldots + a_{m1} \cdot b_{1n} \end{bmatrix}$
the column catenation $X \triangledown Y$, is defined only when $m=m'$ and is given by

\[
\begin{array}{cccccc}
a_{11} & \ldots & a_{1n} & b_{11} & \ldots & b_{1n'} \\
\vdots & \ldots & \ldots & \vdots & \ldots & \vdots \\
\vdots & \ldots & \ldots & \vdots & \ldots & \vdots \\
a_{m1} & \ldots & a_{mn} & b_{m'1} & \ldots & b_{m'n'}
\end{array}
\]

and the row catenation $X \triangle Y$ is defined when $n=n'$ and is given by

\[
\begin{array}{cccccc}
a_{11} & \ldots & a_{1n} \\
\vdots & \ldots & \vdots \\
\vdots & \ldots & \vdots \\
a_{m1} & \ldots & a_{mn} \\
b_{11} & \ldots & b_{1n'} \\
\vdots & \ldots & \vdots \\
\vdots & \ldots & \vdots \\
b_{m'1} & \ldots & b_{m'n'}
\end{array}
\]

**Definition 2.2** If

\[
X = \begin{array}{cccc}
a_{11} & \ldots & a_{1n} \\
\vdots & \ldots & \vdots \\
\vdots & \ldots & \vdots \\
a_{m1} & \ldots & a_{mn}
\end{array}
\]

then the transpose of $X$,

\[
T(X) = \begin{array}{cccc}
a_{11} & \ldots & a_{m1} \\
\vdots & \ldots & \vdots \\
\vdots & \ldots & \vdots \\
a_{1n} & \ldots & a_{mn}
\end{array}
\]

the quarter-turn of $X$ (in the clockwise direction),

\[
QT(X) = \begin{array}{cccc}
a_{m1} & \ldots & a_{1n} \\
\vdots & \ldots & \vdots \\
\vdots & \ldots & \vdots \\
a_{mn} & \ldots & a_{1n}
\end{array}
\]

the reflection about the right-most vertical,

\[
RV(X) = \begin{array}{cccc}
a_{1n} & \ldots & a_{11} \\
\vdots & \ldots & \vdots \\
\vdots & \ldots & \vdots \\
a_{mn} & \ldots & a_{m1}
\end{array}
\]

the reflection about the base,

\[
RB(X) = \begin{array}{cccc}
a_{m1} & \ldots & a_{nn} \\
\vdots & \ldots & \vdots \\
\vdots & \ldots & \vdots \\
a_{1n} & \ldots & a_{11}
\end{array}
\]

and a half-turn,

\[
X^{HT} = \begin{array}{cccc}
a_{mn} & \ldots & a_{m1} \\
\vdots & \ldots & \vdots \\
\vdots & \ldots & \vdots \\
a_{1n} & \ldots & a_{11}
\end{array}
\]

**Definition 2.3** A mapping $H$ from $I^{++}$ to $I'^{++}$ is called homomorphism if $H(X \triangledown Y) = H(X) \triangledown H(Y)$ and $H(X \triangle Y) = H(X) \triangle H(Y)$. It is easily seen that a homomorphism is defined only when $H(a) = r$ by $s$ array of terminals from $I'$, $a$ in $I$, $r$ and $s$ are the same for all $a$ in $I$.

**Definition 2.4** If $X \in \{a,b\}^{++}$, then $X^c$ (the conjugate of $X$) is the matrix in which every $a$ in $X$ is replaced by $b$ and every $b$ by $a$. If $M$ is a set of matrices then $M^c = \{X^c : X \in M\}$.

### III. Definition and Results

**Definition 3.1** A $(R_s : R)$ Array Grammar is a 4-tuple $(V, T, P, S)$ where $V = N \cup I$, $N$ is a finite set of non-terminals, $I$ is a finite set of intermediates, $T$ is a finite set of terminals, $P = P_1 \cup P_2 \cup P_3$, where $P_1$ is a finite set of non-terminal rules, $P_2$ is a finite set of intermediate rules and $P_3$ is a finite set of terminal rules and $S \in N$ is the start symbol. $P_1$ is a finite set of left-linear (or right-linear) rules of the form $u \rightarrow v$, $u \in N$ and $v$ is of the
form $A \diamond B$ (or $B \diamond A$) where $A \in (N \cup I)$ and $B \in I$ and $\diamond$ denotes either row catenation $\triangle$ or column catenation $\nabla$. $P_2$ is the finite set of rules of the form $u \rightarrow v \ u \in I$ and $v = A \diamond B$ (or $B \diamond A$), where $A \in I$ and $B \in T^{++}$, such that each intermediate $K$ in $I$ generates an intermediate regular matrix language $L_k$ using the sequence of elements of $P_2$. $P_3$ is the finite set of rules of the form $E \rightarrow F$, $E \in N$ and $F \in T^{++}$. A language generated by any $(R_s:R)AG$ grammar is denoted by $(R_s:R)AL$.

Derivation proceeds as follows: Starting from $S$, nonterminal rules are applied without any restriction as in a string grammar, till all the nonterminals are replaced, introducing parentheses whenever necessary. Now, replace each intermediate $A$ in $I$, with the elements from the intermediate language $L_A$, subject to the conditions imposed by row and column catenation. The replacements start from innermost parentheses and proceeds outwards. The derivation comes to an end if the condition for row or column catenation is not satisfied.

**Remark 1:** While applying the rules of $P_1$, intermediates are considered as terminals

**Remark 2:** The rules of $P_1$ are in the following forms

- All rules with $\triangle$ are right linear and all rules with $\nabla$ are right-linear.
- All rules with $\triangle$ are left linear and all rules with $\nabla$ are left-linear.

**Proposition 1:** The class of languages generated by $(R_s:R)AG$ is strictly included in the class of languages generated by $(R:R)AG$.

**Proof:**

The language of Staircases of X's

```
.X . . . . . . . . . X
.X . . . . . . . . . X
.X . . . . . . . . . X
.X . . . . . . . . . X
.X . . . . . . . . . X
.X . . . . . . . . . X
.X . . . . . . . . . X
.X X X X X X X X X
```

cannot be generated by the grammar $(R_s:R)AG$ since staircases can be generated through application of both right linear and left linear rules. It is generated by $(R:R)AG$. By the definition of $(R_s:R)AG$, every language generated by it can also be generated by $(R:R)AG$.

**Proposition 2:**

The class of languages generated by $(R_s:R)AG$ is incomparable but not disjoint with the class of languages generated by Regular Matrix Grammars (RMG).

**Proof:**

$L_1 = \left\{ \begin{array}{c}
X . . . . . . . . . X \\
X X . . . . . . . . . X \\
X X X X X X X X X
\end{array} \right\}$

can be generated by $(R_s:R)AG$ but not by RMG.

$L_2 = \left\{ \begin{array}{c}
X . . . . . . . . . X \\
X X X X X X X X X
\end{array} \right\}$

is the set of L tokens of all sizes and all proportions. $L_2$ is a RML but is not a $(R_s:R)AL$.

$L_3$ is the set of all mirror images of L-tokens of all sizes and all proportions i.e.,

$L_3 = \left\{ \begin{array}{c}
X . . . . . . . . . X \\
X X X X X X X X X
\end{array} \right\}$

can be generated by both RML and $(R_s:R)AL$.

**Remark 3:** $(R_s:R)AL$ consists of pictures obtained either by pasting of sub-pictures to left and top (right-linear) or to right and bottom (left-linear).

**Proposition 3:**
The class of languages generated by \((R_s:R)\)AL is closed under Half-turn, Transpose, homomorphism and conjugation operations.

**Proof:**

**Closure under Half-turn:** Since (R:R)AL generates only pictures of fixed proportion and \((R_s:R)\) being a subclass, any picture of the form

\[
P = \begin{pmatrix}
    a_{11} & \cdots & a_{1n} \\
    \vdots & \ddots & \vdots \\
    a_{1n} & \cdots & a_{nn}
\end{pmatrix}
\]

generated by \((R_s:R)\) transformed into

\[
P_{HT} = \begin{pmatrix}
    a_{nn} & \cdots & a_{n1} \\
    \vdots & \ddots & \vdots \\
    a_{1n} & \cdots & a_{11}
\end{pmatrix}
\]

under Half-turn. If \(P\) can be generated by right-linear rules then \(P_{HT}\) can be generated by left-linear rules as follows:

Starting with the base picture at the right bottom and pasting of sub-pictures to its left and top using right linear rules correspond to starting with the base picture at the left top and pasting of sub-pictures to its right and bottom and interchanging the roles of intermediate languages accordingly. Thus both the pictures produced belong to \((R_s:R)\) and hence \((R_s:R)\) is closed under half-turn.

**Closure under transpose:**

**Case(i):** It is to be noted that while transposing a square picture, picture elements which are right to the leading diagonal are transposed to the elements which are below the diagonal and vice-versa and diagonal elements being intact leads to maintaining the structural symmetry of the picture which enables us to choose the basic picture and intermediate languages accordingly to show that \((R_s:R)\) is closed under transpose.

**Case(ii):** When the picture is a non-square one, as in the case of square picture, picture elements which are right to the leading diagonal are transposed to the elements which are below the diagonal and vice-versa and diagonal elements being intact and elements which are not covered in the square process are either in a row (or column) is changed to elements in a column (or row) leads to maintaining structural symmetry of the picture which once again enables us to choose the basic picture and intermediate languages accordingly to show that \((R_s:R)\) is closed under transpose.

Following examples illustrates the above idea: When the picture is a square picture

\[
P = \begin{pmatrix}
    X & X & X \\
    X & X & . \\
    . & X & X
\end{pmatrix}
\]

and its transpose \(P^T = \begin{pmatrix}
    X & X & X \\
    X & X & . \\
    . & X & X
\end{pmatrix}\)

and when the picture is non-square picture

\[
Q = \begin{pmatrix}
    X & X & . \\
    X & . & X \\
    X & X & X
\end{pmatrix}
\]

and its transpose \(Q^T = \begin{pmatrix}
    X & X & . \\
    X & . & X \\
    X & X & X
\end{pmatrix}\)

**Closure under homomorphism and conjugation:**

From the definition of homomorphism and conjugation, it is clear that neither of the operations change the structural symmetry of the picture and hence \((R_s:R)\) is closed under these two operations also.

**Proposition 4:**

The class of languages generated by \((R_s:R)\)AL is not closed under quarter turn, reflection about base and reflection about right-most vertical and row and column catenation. Consider the languages

\[
L_1 = \left\{ \begin{pmatrix}
    X & \cdots & X \\
    X & \cdots & X \\
    X & \cdots & X
\end{pmatrix} \right\}
\]

and
$$L_2 = \left\{ \begin{array}{c} X X X X X X X X \\ X X X X X X X X \\ X X X X X X X X \end{array} \right\}. $$

Then both $L_1$ and $L_2$ belong to $(R_s;R)$ but neither $L_1 \bigtriangleup L_2$ nor $L_1 \bigtriangledown L_2$ belong to $(R_s;R)$ as can be seen from following:

$$L_1 \bigtriangleup L_2 = \left\{ \begin{array}{c} X X X X X X X X \\ X X X X X X X X \\ X X X X X X X X \end{array} \right\}. $$

$$L_1 \bigtriangledown L_2 = \left\{ \begin{array}{c} X X X X X X X X \\ X X X X X X X X \\ X X X X X X X X \end{array} \right\}. $$

Clearly, neither $L_1$ nor $L_2$ is closed under quarter-turn as the quarter-turn pictures of $L_1$ or $L_2$ cannot be generated by only right-linear or only left-linear rules as can be seen from following:

$$\text{QT}(L_1) = \left\{ \begin{array}{c} X X X X X X X X \\ X X X X X X X X \\ X X X X X X X X \end{array} \right\} $$

and $\text{QT}(L_2) = \left\{ \begin{array}{c} X X X X X X X X \\ X X X X X X X X \\ X X X X X X X X \end{array} \right\}$. 

Hence $(R_s;R)$ is not closed under quarter-turn.

Following counter examples show that $(R_s;R)$ is not closed under reflection about base and reflection about vertical operations.

$$R_B(L_1) = \left\{ \begin{array}{c} X X X X X X X X \\ X X X X X X X X \\ X X X X X X X X \end{array} \right\} $$

and pictures of the form and pictures of the form

$$R_V(L_1) = \left\{ \begin{array}{c} X X X X X X X X \\ X X X X X X X X \\ X X X X X X X X \end{array} \right\} .$$

IV. CONCLUSION

In this paper, we have compared $(R_s;R)$AL with RMG languages[1] and $(R;R)$AL[2].

REFERENCES


Received: August 10, 2015; Accepted: September 30, 2015

UNIVERSITY PRESS
Website: http://www.malayajournal.org/