Two-dimensional Picture Grammar models

K.G. Subramanian
School of Mathematical Sciences,
Universiti Sains Malaysia,
11800 Penang, Malaysia

M. Geethalakshmi
Department of Mathematics
Dr. MGR Janaki College
Chennai , India

Atulya K. Nagar
Deanery of Business and Computer Sciences,
Liverpool Hope University,
Hope Park, Liverpool L16 9JD UK

S.K. Lee
School of Mathematical Sciences,
Universiti Sains Malaysia,
11800 Penang, Malaysia

Abstract

A new theoretical model of grammatical picture generation called extended 2D context-free picture grammar (E2DCFPG) generating rectangular picture arrays of symbols is introduced. This model which allows variables in the grammar and uses the squeezing mechanism of forming the picture language over terminal symbols, is an extension of the pure 2D context-free picture grammar (P2DCFPG) [13]. The extended picture grammar model E2DCFPG is shown to have more picture generative power than the P2DCFPG and certain other existing 2D models. Certain closure and other properties of this new model are also examined.

1. Introduction

Syntactic models of picture generation have become established as one of the main areas of research in theoretical studies on digital pictures and Image analysis. A variety of picture language generating models have been introduced in the literature such as the two-dimensional matrix grammars [10], array grammars [7, 8, 14], chain-code picture grammars [5], tiling systems [2, 3] and so on. Most of these grammars utilize the techniques and results of the rich theory of formal string languages.

A two-dimensional grammar model, called pure 2D context-free picture grammar (P2DCFPG), for picture array generation was introduced in [13] based on pure context-free (string) grammars [6] that make use of only terminal symbols. In this paper we allow variables in the rules of this picture grammar model and collect the picture arrays generated over a set of terminal symbols. The resulting two-dimensional grammar model is called extended 2D context-free picture grammar. Unlike the models in [10, 11], rewriting any column or any row of the rectangular array is allowed in this model and no priority of rewriting is prescribed as in [10] in which the second phase of generation can take place only after the first phase is over. The extended picture grammar model E2DCFPG is shown to have more picture generative power than the P2DCFPG and certain other existing 2D models. Certain closure and other properties of this new model are also examined.

2. Preliminaries

Let Σ be a finite alphabet. A word or string $w = a_1a_2\dots a_n \quad (n \geq 1)$ over Σ is a sequence of symbols from Σ. The length of a word w is denoted by $|w|$. The set of all words over Σ, including the empty word λ with no symbols, is denoted by Σ^*. We call words of Σ^* as horizontal words. For any word $w = a_1a_2\dots a_n$, we denote by w^T the vertical word

$$\begin{matrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{matrix}$$

We also define $(w^T)^T = w$. We set λ^T as λ itself. A rectangular $m \times n$ array M over Σ is of the form

$$M = \begin{matrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{matrix}$$

where each $a_{ij} \in \Sigma, 1 \leq i \leq m,\ 1 \leq j \leq n$. The set of all rectangular arrays over Σ is denoted by Σ^{**}, which includes the empty array λ. $\Sigma^{**} - \{\lambda\} = \Sigma^{++}$. We denote respectively by $\circ$ and $\diamond$ the *column concatenation* and *row*

978-0-7695-3325-4/08 $25.00 © 2008 IEEE
DOI 10.1109/EMS.2008.72

concatenation of arrays in Σ^{**}. In contrast to concatenation of strings, these operations are partially defined, namely, for any $X, Y \in \Sigma^{**}$, $X \circ Y$ is defined if and only if X and Y have the same number of rows. Similarly $X \diamond Y$ is defined if and only if X and Y have the same number of columns.

We refer to [7, 8] for array grammars. For notions of formal language theory we refer to [9]. We briefly recall some of the picture generating models that are needed in the subsequent section.

In the 2D grammar model introduced in [10], which we call as Siromoney regular or context-free matrix grammar (SRMG or SCFMG), a horizontal word $S_{i1} \ldots S_{in}$ over intermediate symbols is generated by a Chomskian grammar. Then simultaneously from each intermediate symbol S_{ij} a vertical word of the same length over terminal symbols is derived to constitute the jth column of the rectangular array generated. We denote the picture language classes of Siromoney regular, context-free matrix grammars by RML, CFML respectively.

The Siromoney regular / context-free matrix grammars were extended in [12] by specifying a finite set of tables of rules in the second phase of generation with each table having either right-linear nonterminal rules or right-linear terminal rules. At one step of derivation, the rules in a table are used for rewriting.The resulting families of picture array languages are denoted by TRML and TCFML and are known to properly include RML and CFML respectively.

In the extended tabled 0L array system (ET0LAS) G introduced in [11] for generating rectangular picture arrays, a derivation in G takes place as follows: Starting with a rectangular array $M_1 \in \Sigma^{++}$, all the symbols of either the rightmost or leftmost column or the uppermost or lowermost row of M_1 are rewritten in parallel, respectively by the rules of a left or a right table or an up or a down table to yield a rectangular array M_2. A set $\mathcal{M}(G)$ of rectangular arrays is called an extended tabled $0L$ array language (ET0LAL) if and only if there exists an extended tabled $0L$ array system G with the set T of terminal symbols such that $\mathcal{M}(G) = \{M | M_0 \Rightarrow^* M, M \in T^{**}\}$. The family of extended tabled $0L$ array languages is also denoted by ET0LAL itself.

Based on a well known characterization of recognizable string languages in terms of local languages and projections, an interesting model of tiling recognizable languages describing rectangular picture arrays was introduced in [2, 3].The family of local picture array languages is denoted by LOC.The family of tiling recognizable picture array languages is denoted by REC.

3. Extended 2D Picture Grammars

An extended 2D context-free picture grammar (E2DCFPG) is a 5-tuple $G = (V, \Sigma, P_c, P_r, \mathcal{M})$ where

- V is a finite set of symbols ; The elements of $V - \Sigma$ are called variables;
- $\Sigma \subseteq V$ is the set of terminal symbols;
- $P_c = \{t_{c_i}/1 \le i \le m\}, P_r = \{t_{r_j}/1 \le j \le n\}$;

Each $t_{c_i}, (1 \le i \le m)$, called a column table, is a set of context-free rules of the form $A \to \alpha, A \in V - \Sigma, \alpha \in V^*$ such that for any two rules $A \to \alpha, B \to \beta$ in t_{c_i}, we have $|\alpha| = |\beta|$; Each $t_{r_j}, (1 \le j \le n)$, called a row table, is a set of context-free rules of the form $C \to \gamma^T, C \in V - \Sigma$ and $\gamma \in V^*$ such that for any two rules $C \to \gamma^T, D \to \delta^T$ in t_{r_j}, we have $|\gamma| = |\delta|$;

- $\mathcal{M} \subseteq V^{**} - \{\lambda\}$ is a finite set of axiom arrays.

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 every symbol of a column of M_1 by rules of some column table t_{c_i} in P_c or of a row of M_1 by rules of some row table t_{r_j} in P_r. The reflexive transitive closure of $\Rightarrow$ is denoted by $\Rightarrow^*$.

The picture array language $L(G)$ generated by G is the set of rectangular picture arrays $\{M/M_0 \Rightarrow^* M \in \Sigma^{**}$, for some $M_0 \in \mathcal{M}\}$. The family of picture array languages generated by extended 2D context-free picture grammars is denoted by E2DCFPL.

Example 1 Consider the E2DCFPG $G = (V, \Sigma, P_c, P_r, \{M_0\}), V = \{S, X, Y, Z, A, a, b\}, \Sigma = \{a, b\}, P_c = \{t_{c_1}, t_{c_2}\}, P_r = \{t_{r_1}, t_{r_2}\}, t_{c_1} = \{S \to aSa, Y \to BZB, X \to bXb\}, t_{c_2} = \{S \to a, Y \to A, X \to a\}$, $t_{r_1} = \{B \to b, A \to a\}$,, $t_{r_2} = \left\{B \to \begin{smallmatrix} B \\ b \end{smallmatrix}, Z \to \begin{smallmatrix} Y \\ X \end{smallmatrix}\right\}, M_0 = \begin{smallmatrix} a & S & a \\ B & Y & B \end{smallmatrix}$

G generates picture arrays M of the form in Figure 1.

Remark 1. When the symbol 'b' is interpreted as a blank, the array M represents the digitized form of the English letter T with the body of the letter being made of $a's$. (Figure 2). Note also that all the three 'arms' of T are of equal length.

2. In the definition of E2DCFPG, we can take a single nonterminal start symbol S and allow a special start table consisting of the rule $S \to M_0$ for every axiom array M_0 in the grammar. We can require the start symbol S not to occur in the axiom arrays or in any rule in the column or row tables. This modification will give rise to an equivalent grammar generating the same picture language as the given E2DCFPG.

4. Comparisons and Closure results

We now compare the new extended 2D picture grammar model introduced here with those in [13, 10, 11, 2, 3].

Theorem 1 The family of E2DCFPL properly contains the family of P2DCFPL [13].

$$M_0 \Rightarrow \begin{array}{ccccc} a & a & S & a & a \\ B & B & Z & B & B \end{array} \Rightarrow \begin{array}{ccccc} a & a & S & a & a \\ B & B & Y & B & B \\ b & b & X & b & b \end{array}$$

$$\Rightarrow \begin{array}{ccccccc} a & a & a & S & a & a & a \\ B & B & B & Z & B & B & B \\ b & b & b & X & b & b & b \end{array}$$

$$\Rightarrow \begin{array}{ccccccc} a & a & a & S & a & a & a \\ B & B & B & Y & B & B & B \\ b & b & b & X & b & b & b \\ b & b & b & X & b & b & b \end{array}$$

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

$$\Rightarrow \begin{array}{ccccccc} a & a & a & a & a & a & a \\ b & b & b & a & b & b & b \\ b & b & b & a & b & b & b \\ b & b & b & a & b & b & b \end{array} = M$$

Figure 1. Derivation of a picture array M

$$\begin{array}{ccccccc} a & a & a & a & a & a & a \\ & & & a & & & \\ & & & a & & & \\ & & & a & & & \end{array}$$

Figure 2. Digitized Letter T

Proof. The containment can be seen as follows: Corresponding to a P2DCFPG G generating the given P2DCFPL L, we form a E2DCFPG to generate L. For every terminal symbol of the P2DCFPG G, create a distinct nonterminal and then replace every (terminal) symbol in the axiom of G and in the left and right sides of the rules of the tables of G by the corresponding nonterminal symbol created. Add a table of rules of the form $A \rightarrow a$ where A is the new nonterminal created corresponding to the terminal symbol a. The resulting set of tables constitute the tables of rules of the E2DCFPG to be formed. It can be seen that L is generated by the E2DCFPG formed. For proper containment the picture array language of Example 1 can not be generated by any P2DCFPG. In fact to 'grow' the horizontal arm of the array representing the letter T, a rule of the form $a \rightarrow aaa$ is needed but then this rule could be applied to any a in the vertical column of $a's$ yielding picture arrays not in the language.

Theorem 2 The family of E2DCFPL properly contains the families of RML and CFML [10].

Proof. To prove the containment, we note that in constructing a E2DCFPG G to generate the given RML or CFML generated respectively by a corresponding SRMG [10] or SCFMG [10], a column table of G is formed consisting of the rules of the SRMG or SCFMG and two row tables are formed, one consisting of the nonterminal right-linear rules of the form $A \rightarrow aB$ and the other, the terminal rules of the form $A \rightarrow a$ (A, B are nonterminals and a is a terminal) of the second phase of the SRMG or SCFMG. Proper containment follows from the fact that the picture language in example 1 cannot be generated by any Siromoney context-free matrix grammar and hence by any Siromoney regular matrix grammar since the two phases of derivation are independent in such a grammar whereas each of the generated pictures of example 1, has an equal number of columns to the left and right of the middle column of $a's$ which equals the number of rows below the first row of $a's$.

Theorem 3 The family of E2DCFPL properly contains the families of TRML and TCFML [12].

Proof. The proof of containment is similar to Theorem 2 except that the the tables of rules in the second phase of the 2D grammars TRMG [12] and TCFMG [12] of the corresponding TRML and TCFML are taken as the row tables of the E2DCFPG.The proper containment follows from the same example 1 as even in a TRMG or a TCFMG the two phases of derivation are independent.

Remark In [13], it is shown that the class P2DCFPL is incomparable but not disjoint with all the four families of RML, CFML, TRML and TCFML in contrast to the proper containment result in Theorems 2 and 3, therby showing the increase in generative power of the extended 2D CF picture grammar model considered here although this increase in generative power is to be expected.

But the salient feature of P2DCFPL, namely, taking care of the shearing effect in replacing a subarray of a given rectangular array by rewriting a row or column of symbols in parallel by equal length strings, is present in this E2DCFPL as well. Also this new model is related to the ET0LAS in [11] in the sense that a column or row of symbols of a rectangular array is rewritten in parallel but the difference is that the rewriting is done only at the "edges" of a rectangular array in an ET0LAS whereas here we allow rewriting in parallel of any column or row of symbols.

Theorem 4 The family of E2DCFPL properly contains the family of ET0LAL [11] and hence properly contains the family of T0LAL [11].

Proof. To prove the containment, for every nonterminal and terminal symbol α in the ET0LAS G_1 generating the given ET0LAL, we create two new nonterminals

α_1, α_2 in the E2DCFPG G_2 to be formed. The symbols in the 'border' of the axiom arrays in G_1 (it could be a single symbol also) are replaced by the new nonterminal of type α_1 except that the 'four' corners are replaced by the new nonterminal of type α_2 (suitable modifications can be done for the degenerate cases of only two corners or a single corner) and these arrays are taken as the axiom arrays in G_2. The right and left tables in G_1 (used for rewriting the rightmost or leftmost column of an array in derivations in G_1), are the column tables of G_2 with the following modification in the rules: Every rule of the form $X \rightarrow AB \dots CZ$, in a right table is replaced by two rules of the form $X_1 \rightarrow AB \dots CZ_1$ (only the last symbol in the right is changed) and $X_2 \rightarrow A_1 B_1 \dots C_1 Z_2$ (every symbol in the right is changed) where $X_1, A_1, B_1, \dots Z_1$, are α_1 kind of new nonterminals and X_2, Z_2 are α_2 kind of new nonterminals. Note that the rule of the type $X_1 \rightarrow AB \dots CZ_1$ will rewrite an 'inner' symbol in the rightmost column of the array rewritten whereas the rule of the type $X_2 \rightarrow A_1 B_1 \dots C_1 Z_2$ will rewrite a 'corner' symbol. Similar column table of rules is formed corresponding to a left table in G_1. Likewise for up and down tables of G_1. Finally, for each table in the ET0LAS G_1 consisting of rules of the form $A \rightarrow \alpha, \alpha$ is entirely made of terminal symbols, we add a table of rules of the form $A_1 \rightarrow \alpha, A_2 \rightarrow \alpha$ where A_1, A_2 are the two kinds of nonterminals introduced in G_2 correponding to the nonterminal A in G_1. It can be seen that G_2 generates the ET0LAL given.

For proper containment we note that in a ET0LAS the 'growth' can take place only in a border whereas an inner column or row cannot be grown or the terminal symbols in the inner part cannot be altered. So we can construct picture languages that are in E2DCFPL but not generated by any ET0LAS. One such picture language is L over the terminal symbols a, b, c, d consisting of rectangular arrays of size $p \times 2(n+m+1)$, $p \geq 1, n, m \geq 0$ and of the form $(a^n c a^n)_p (b^m d b^m)_p$ where $(w)_p$ for a word w denotes a rectangular array with p rows, each row being w. Note that in any row in the array the symbols between a c and a d "grow" which is not possible in an ET0LAS.

Theorem 5 E2DCFPL - REC [2] $\neq \phi$. In particular E2DCFPL - LOC [2] $\neq \phi$.

Proof. The picture array language L consisting of arrays $M = M_1 \circ c \circ M_1$ where M_1 is a string over a (M is a picture array with only one row) is in the class E2DCFPL. In fact it is generated by a E2DCFPG with a column table consisting of the rule $C \rightarrow aCa$ and another table consisting of the rule $C \rightarrow c$. But L is known [2] to be not in REC and hence not in LOC.

Theorem 6 The family of E2DCFPL is closed under union, transposition, reflection about the base and reflection about the leg but not under column catenation or row catenation.

Proof. If L_1, L_2 are two E2DCFPL generated respectively by the E2DCFPG G_1, G_2 with distinct nonterminals. Closure under union can be seen by creating a E2DCFPG G to generate $L_1 \cup L_2$ such that the axiom in G is a new symbol S and having a special start table of rules of the form $S \rightarrow M_0$ for every axiom array M_0 in G, in addition to retaining the tables of rules of G_1 and G_2.

If L is a picture array language generated by a E2DCFPG G and L^T is the transposition of L, then the P2DCFG G' to generate L^T is formed by taking the column tables of G as row tables and row tables as column tables but for a rule $A \rightarrow \alpha$ in a column table of G, the rule $A \rightarrow \alpha^T$ is added in the corresponding row table of G' and likewise for a rule $B \rightarrow \beta^T$ in a row table of G, the rule $B \rightarrow \beta$ is added in the corresponding column table of G'. Closure under the operations of reflection about base, reflection about leg are similar.

Non-closure under column catenation is seen by considering two picture languages L_1, L_2 over the terminal symbols a, b such that the arrays in L_1 are of the form $(a^n)_{2p} \diamond b^n \diamond (a^n)_p, n, p \geq 1$ and the arrays in L_2 are of the form $(a^n)_p \diamond b^n \diamond (a^n)_{2p}, n, p \geq 1$. When any two arrays are column catenated, in the array obtained the $b's$ will be in a row in the 'left half' with the rows of $a's$ above and below in the ratio of 1:2 whereas the $b's$ will be in a different row in the 'right half' with the rows of $a's$ above and below in the ratio of 2:1, so that this feature cannot be handled by any E2DCFPG. Non-closure under row catenation is similar.

5. Conclusion

The two-dimensional picture array generating model E2DCFPG introduced here is more general than the ET0LAS in [11]. A study on controlling the application of the tables as done in [11] by a control language might further increase the generative power and thus might help describe more complex picture languages. The idea of interpreting letter symbols in a picture array by primitive patterns is a well-known technique to obtain interesting classes of "kolam" [10] pictures or chain code [5] pictures and so on. We can employ this technique to generate such picture patterns as an application of the extended 2D CF picture grammars.There are many application areas of two-dimensional array grammars such as character recognition [1], region filling algorithms [4] and so on. It remains to explore the capability of the theoretical model proposed here for such application problems.

6. Acknowledgement

The authors are grateful to the referees for useful comments. The authors K.G. Subramanian and S.K. Lee gratefully acknowledge support for this research by FRGS Grant of the Government of Malaysia.

References

[1] H. Fernau and R. Freund, Bounded parallelism in array grammars used for character recognition, *Lecture notes in Computer science*, Springer-Berlin, **1121**, 1996, 40-49.

[2] D. Giammarresi, A. Restivo, Two-dimensional languages, In *"Handbook of Formal Languages"* Vol.**3**, Eds. G. Rozenberg and A. Salomaa, Springer Verlag, 1997, 215 - 267.

[3] D. Giammarresi, A. Restivo, Recognizable Picture Languages, *International Journal of Pattern Recognition and artificial Intelligence*, Special issue on Parallel Image Processing, Eds. M. Nivat, A. Saoudi and P.S.P. Wang, 1992, 31-46.

[4] E.T. Lee, Y.J. Pan and P. Chu, An algorithm for region filling using two-dimensional grammars, *Int. Journal of Intelligence Systems*, **2**, 255-263.

[5] H.A. Maurer, G. Rozenberg, E. Welzl, Chain-code picture languages, *Lecture notes in Computer science*, Springer-Berlin, **153**, 1983, 232-244.

[6] H.A. Maurer, A. Salomaa, D. Wood, Pure Grammars, *Information and Control*, **44**, 1980, 47-72.

[7] A. Rosenfeld, *Picture Languages - Formal Models for Picture Recognition*, Academic Press, New York, 1979.

[8] A. Rosenfeld and R. Siromoney, Picture languages - a survey, *Languages of design*, **1**, 1993, 229–245.

[9] A. Salomaa, Formal languages, Academic Press, London, 1973.

[10] G. Siromoney, R. Siromoney, K. Krithivasan, Abstract families of matrices and picture languages, *Computer Graphics and Image Processing*, **1**, 1972, 234-307.

[11] R. Siromoney and G. Siromoney, Extended Controlled Tabled L- arrays, *Information and Control*, **35(2)**, 1977, 119-138.

[12] R. Siromoney, K.G. Subramanian, K. Rangarajan, Parallel/Sequential rectangular arrays with tables, *International Journal of Computer Mathematics*, **6A**, 1977, 143-158.

[13] K.G. Subramanian, A.K. Nagar, M. Geethalakshmi, Pure 2D Picture Grammars (P2DPG) and P2DPG with Regular Control, Lecture Notes in Computer Science 4958, Springer-verlag Berlin Heidelberg, 2008, 330-341.

[14] P.S.P. Wang, *Array grammars, Patterns and recognizers*, World Scientific, 1989.