INFORMATION AND CONTROL 20, 143–149 (1972)

Matrix Grammars with a Leftmost Restriction

ARTO SALOMAA

Department of Mathematics, University of Turku, Turku, Finland

The family of languages generated by matrix grammars with context-free (context-free λ-free) core productions and with a leftmost restriction on derivations equals the family of recursively enumerable (context-sensitive) languages.

INTRODUCTION

Matrix grammars introduced by Ábrahám (1965) have proved to be a very fruitful generalization of context-free grammars: simple in principle, easy to deal with and yet very powerful [cf. Brainerd (1968), Ibarra (1970) and Siromoney (1969)]. They fall into the category of grammars with restricted use of productions, and possess the same generative capacity as programmed grammars, periodically time-variant grammars and grammars with a regular control language [cf. Salomaa (1970)].

In this paper, we consider matrix grammars, where the core productions in the matrices have the context-free form $X \to P$, X being a nonterminal. It is known [cf. Rosenkrantz (1969) and Salomaa (1970)] that if the core productions are also λ-free, i.e., the right side P is always distinct from the empty word λ, then the family of languages generated by such matrix grammars is properly included in the family of context-sensitive languages. On the other hand, if λ is allowed on the right side and, furthermore, an appearance-checking interpretation in the application of productions is considered, then the family of generated languages coincides with the family of recursively enumerable languages [Rosenkrantz (1969) and Salomaa (1970)]. (In this interpretation, a production $X \to P$ may be applied by (i) noticing that X does not occur in the word under scan, and (ii) moving on to the next production.) It is an open problem how large the family of generated languages will be if λ is allowed on the right side of productions but the appearance-checking interpretation is not considered. For instance, it is not known whether nonregular languages over one letter belong to this family [cf. Salomaa (1970) and (1970a)].

In this paper we introduce a restriction on the application of matrices: before applying a matrix m to a word Q such that the application of m begins with rewriting the i-th letter of Q, one has to make sure that no matrix m' is applicable to Q such that the application of m' begins with rewriting the j-th letter of Q, where $j < i$. This "leftmost restriction" concerns only the first productions in the matrices.

It turns out that the family of generated languages equals the family of context-sensitive languages or the family of recursively enumerable languages, depending on whether or not the core productions are assumed to be λ-free. This establishes the interconnection between matrix grammars and the basic Chomsky hierarchy of language families, and gives another characterization of two of the families in this hierarchy.

1. Definitions and Results

A *matrix grammar* is an ordered quadruple

$$G = (V_N, V_T, X_0, M),$$

where V_N and V_T are disjoint alphabets (*nonterminal* and *terminal* alphabet), $X_0 \in V_N$ (*initial* letter), and M is a finite set of finite sequences whose elements are ordered pairs (X, P) such that $X \in V_N$ and P is a word over the alphabet $V = V_N \cup V_T$. The ordered pairs (X, P) are called *productions* and written $X \to P$. Thus, the elements m of M are finite sequences of productions. They are written

$$m = [X_1 \to P_1, \dots, X_r \to P_r], \qquad r \geqslant 1, \tag{1}$$

and referred to as *matrices*.

A binary relation $\Rightarrow$ on the set $W(V)$ of all words over V is defined as follows. $Q \Rightarrow R$ holds iff there exist an integer $r \geqslant 1$, words

$$Q_1, \dots, Q_{r+1}, \quad P_1, \dots, P_r, \quad R_1, \dots, R_r, \quad R^1, \dots, R^r \tag{2}$$

over V and letters $X_1, \dots, X_r$ of V_N such that (i) $Q_1 = Q$ and $Q_{r+1} = R$, (ii) the matrix (1) is in M, and (iii) $Q_i = R_i X_i R^i$ and $Q_{i+1} = R_i P_i R^i$, for every $i = 1, \dots, r$. If (i)–(iii) are satisfied, we also say that $Q \Rightarrow R$ holds with *specifications* (m, R_1).

The length of a word P is denoted by $\lg(P)$. By definition, $\lg(\lambda) = 0$.

A binary relation $\Rightarrow_{\text{left}}$ on the set $W(V)$ is defined as follows. $Q \Rightarrow_{\text{left}} R$

holds iff (i) For some m and R_1, $Q \Rightarrow R$ holds with specifications (m, R_1), and (ii) For no m', R' and R_1' such that $\lg(R_1') < \lg(R_1)$, $Q \Rightarrow R'$ holds with specifications (m', R_1'). Let $\overset{*}{\Rightarrow}_{\text{left}}$ be the reflexive transitive closure of the relation $\Rightarrow_{\text{left}}$. The language *generated* by the matrix grammar G under *leftmost restriction* on derivations is defined by

$$L_{\text{left}}(G) = \{P \in W(V_T) \mid X_0 \overset{*}{\Rightarrow}_{\text{left}} P\}.$$

THEOREM 1. *A language L is recursively enumerable iff there exists a matrix grammar G such that* $L = L_{\text{left}}(G)$.

Consider *state grammars* introduced by Kasai (1970). Modify this notion by allowing the empty word λ to appear in state productions. (This means that, in the definition of a state grammar in Kasai (1970), V^+ is replaced by V^*.) It is a consequence of Theorem 1 that state languages thus defined coincide with recursively enumerable languages.

A matrix grammar G is termed *λ-free* iff all words P_i in every matrix (1) are distinct from the empty word λ.

THEOREM 2. *A language L is context-sensitive iff there exists a λ-free matrix grammar G such that* $L = L_{\text{left}}(G)$.

2. PROOFS

We will first prove Theorem 1. It is obvious that, for any matrix grammar G, the language $L_{\text{left}}(G)$ is recursively enumerable. The converse follows from Theorem 2 and the fact that every recursively enumerable language is obtained from a context-sensitive language by erasing some letters. However, we will give also a proof which does not use these facts.

Let G be a matrix grammar with the set of productions F and let F_1 be a subset of F. A binary relation $\Rightarrow_c$ on the set $W(V)$ is defined as follows (V has the same meaning as in Section 1): $Q \Rightarrow_c R$ holds iff there exist an integer $r \geqslant 1$, words (2) over V and letters $X_1, \ldots, X_r$ of V_N such that (i) $Q_1 = Q$ and $Q_{r+1} = R$, (ii) the matrix (1) is a matrix of G, and (iii) for each $i = 1, \ldots, r$, either $Q_i = R_i X_i R^i$ and $Q_{i+1} = R_i P_i R^i$, or else $X_i \to P_i$ belongs to F_1, $Q_i = Q_{i+1}$ and X_i does not appear in Q_i. Let $\overset{*}{\Rightarrow}_c$ be the reflexive transitive closure of the relation $\Rightarrow_c$. (Note that this is the appearance-checking interpretation in the application of productions mentioned in the Introduction.)

Let L be a recursively enumerable language. By Lemma 2 in Salomaa

(1970), there is a matrix grammar G_1 and a subset F_1 of the production set of G_1 such that

$$L = \{P \in W(V_T) \mid X_0 s_0 \overset{*}{\Rightarrow}_c P\},$$

where X_0 and s_0 are nonterminals, and V_T the terminal alphabet of G_1. Moreover, the matrices of G_1 are of the form

$$[f, s \to s'], \tag{3}$$

where f is a context-free production and s' is a single nonterminal or λ. Furthermore, none of the nonterminals s and s' appearing in the second productions of (3) appears in the first productions f, and F_1 is a subset of the set of the first productions f.

We will define a matrix grammar G_2 such that

$$L = L_{\text{left}}(G_2). \tag{4}$$

Let $V_N(S)$ be the set of those nonterminals of G_1 which appear in the first (second) productions of the matrices (3). Define

$$V_N' = \{Y' \mid Y \in V_N\}, \qquad V_N'' = \{Y'' \mid Y \in V_N\}.$$

Let V_1 be the set consisting of the nonterminals U, U'', Z_0, Z_1, Z_2 and Z_f, where f ranges over F_1. These nonterminals are assumed to be distinct from the ones previously introduced. For a word P over $V_N \cup V_T$, we denote by P'' the word obtained from P by replacing the rightmost nonterminal Y with Y'' and the other nonterminals Y with Y'. If P is a word over V_T, then P'' is defined to be the word PU''.

The nonterminal alphabet of G_2 is the union

$$V_N \cup S \cup V_N' \cup V_N'' \cup V_1,$$

Z_0 being the initial letter and V_T the terminal alphabet. We now define the matrices of G_2. Consider an arbitrary matrix (3) of G_1, and assume that f stands for the production

$$X \to P, \qquad X \in V_N, \qquad P \in W(V_N \cup V_T).$$

For each such matrix, G_2 contains all of the following matrices:

$$\begin{array}{ll}
[Y \to Y', Z_1 \to Z_1] & \text{for each} \quad Y \in V_N, \\
[X \to P'', Z_1 \to Z_2, s \to s'], & \\
[Y' \to Y, Z_2 \to Z_2] & \text{for each} \quad Y \in V_N, \\
[Y'' \to Y, Z_2 \to Z_1] & \text{for each} \quad Y \in V_N, \\
[U'' \to \lambda, Z_2 \to Z_1]. &
\end{array}$$

If f belongs to the set F_1, G_2 contains the additional matrices

$$[Z_1 \to Z_f, s \to s'], [X \to U, Z_f \to Z_f], [X' \to U, Z_f \to Z_f], [Z_f \to Z_1]. \quad (5)$$

Furthermore, G_2 contains the matrices

$$[Z_0 \to X_0 Z_1 s_0], \qquad [Z_1 \to \lambda].$$

It can now be verified that (4) is true. Derivations according to G_2 begin from the word $X_0 Z_1 s_0$, and simulate the derivations according to G_1. At first, nonterminals are marked with primes, until a suitable occurrence of X is found. Then the rewriting according to f, as well as the corresponding change between the s's, are performed, after which all primes are removed. The nonterminal marked with a double prime is the rightmost nonterminal carrying primes and, hence, all primes have been removed when the double prime is removed. If no occurrence of X is found, then the change between the s's is performed by (5), after which the primes are removed. Finally, Z_1 may be removed. This proves the inclusion $L \subseteq L_{\text{left}}(G_2)$. The reverse inclusion follows because the Z's rule out the possibility of other derivations leading to terminal words. This completes the proof of Theorem 1.

To prove Theorem 2, we assume first that G is a λ-free matrix grammar. To avoid tedious (and straightforward) constructions, we only give an informal description of a context-sensitive grammar G_1 generating the language $L_{\text{left}}(G)$. We note first that each matrix (1) of G determines a unique minimal set T of nonterminals which have to be present in a word Q in order for (1) to be applicable to Q. For instance, for the matrix

$$[X_1 \to X_1 X_2 a, X_3 \to X_1 X_3, X_2 \to ab],$$

we have $T = \{X_1, X_3\}$. For each such T obtained from the matrices of G, G_1 contains the nonterminal Y^T. Before an intended application of a matrix (1) to a word Q, beginning with a particular occurrence of X_1, a marker Y is placed in front of that occurrence of X_1. The nonterminals Y^T travel across the word Q, checking that the leftmost restriction is satisfied. The actual application of (1) can clearly be carried out using context-sensitive productions. Instead of the grammar G_1, one may introduce a linear bounded automaton.

Conversely, assume that L is a context-sensitive language. (We assume that L does not contain the empty word λ.) It is well known [e.g., cf. Salomaa (1969), Theorem IV.6.5] that L is generated by a grammar

$G_1 = (V_N, V_T, X_0, F)$, where all productions in F are of the three forms

$$X \to YZ, \qquad X, Y, Z \in V_N, \tag{6}$$

$$f: XU \to YZ, \qquad X, U, Y, Z \in V_N, \tag{7}$$

$$X \to a, \qquad X \in V_N, \quad a \in V_T. \tag{8}$$

Let V_N' and V_N'' be defined as above, and let V_1 consist of the nonterminals $A_0, A_1, A_2, A_3, A_4, A_f$, where f ranges over the productions (7) in F. Consider the matrix grammar

$$G_2 = (V_N \cup V_N' \cup V_N'' \cup V_1, V_T \cup \{\#\}, A_0, M),$$

where M consists of the matrices

$$\begin{array}{ll}
[A_0 \to X_0 A_1], & \\
[Y \to Y', A_1 \to A_1] & \text{for each } Y \in V_N, \\
[X \to Y'Z'', A_1 \to A_2] & \text{for each production (6) in } F, \\
[X \to Y', A_1 \to A_f] & \text{for each production (7) in } F, \\
[U \to Z'', A_f \to A_2] & \text{for each production (7) in } F, \\
[B \to A_4, A_f \to A_f] & \text{for each production (7) in } F \\
 & \quad \text{and each } B \neq U, B \in V_N, \\
[Y' \to Y, A_2 \to A_2] & \text{for each } Y \in V_N, \\
[Y'' \to Y, A_2 \to A_1] & \text{for each } Y \in V_N, \\
[X \to a, A_1 \to A_3] & \text{for each production (8) in } F, \\
[X \to a, A_3 \to A_3] & \text{for each production (8) in } F, \\
[X \to a, A_3 \to \#] & \text{for each production (8) in } F, \\
[X \to a, A_1 \to \#] & \text{for each production (8) in } F.
\end{array}$$

It is now easy to verify that

$$L_{\text{left}}(G_2) = L\{\#\}. \tag{9}$$

(Note that A_4 can never be eliminated. It will be introduced if one tries to simulate an application of (7) to an occurrence of U which is not the next letter to the right of an occurrence of X.)

Thus, we have shown that, for every context-sensitive language L, there is a matrix grammar G_2 such that (9) is satisfied. In (9) the letter $\#$ may be replaced by any fixed terminal letter, and our result still holds. This additional letter can be eliminated as follows. Every context-sensitive language (in fact, every language not containing the empty word) over the alphabet V_T can be expressed in the form

$$L_1 = \bigcup_{a \in V_T} L_1^{(a)}\{a\} \cup L_2 , \tag{10}$$

where

$$L_1^{(a)} = \{P \mid Pa \in L_1 \text{ and } P \neq \lambda\},$$

and L_2 consists of all letters a which are in L_1. Since L_1 is context-sensitive, each of the languages $L_1^{(a)}$ is context-sensitive. Using the result established above for L, we see that each member of the union (10) is of the form $L_{\text{left}}(G)$, where G is a matrix grammar. Since languages of this form are obviously closed under union, L_1 itself is of this form, which completes the proof of Theorem 2.

Acknowledgment

The author is grateful to the referee for several useful remarks.

Received: February 1, 1971

References

Ábrahám, S. (1965), Some questions of phrase structure grammars, *Computational Linguistics* **4**, 61–70.

Brainerd, B. (1968), An analog of a theorem about context-free languages, *Information and Control* **11**, 561–567.

Ibarra, O. (1970), Simple matrix languages, *Information and Control* **17**, 359–394.

Kasai, T. (1970), An hierarchy between context-free and context-sensitive languages, *J. Comput. System Sci.* **4**, 492–508.

Rosenkrantz, D. J. (1969), Programmed grammars and classes of formal languages, *J. Assoc. Comput. Mach.* **16**, 107–131.

Salomaa, A. (1970), Periodically time-variant context-free grammars, *Information and Control* **17**, 294–311.

Salomaa, A. (1970a), On some families of formal languages obtained by regulated derivations, *Ann. Acad. Sci. Fenn. Ser. A* **I**, 479.

Salomaa, A. (1969), "Theory of Automata," Pergamon Press, New York.

Siromoney, R. (1969), On equal matrix languages, *Information and Control* **14**, 135–151.