KYBERNETIKA — VOLUME *14* (1978), NUMBER 6

Two Infinite Hierarchies of Languages Defined by Branching Grammars

GHEORGHE PĂUN

Two types of branching grammars are introduced inspired from Havel's works about branching automata. They naturally define two infinite hierarchies into the family of regular, respectively, linear languages. Homomorphic characterisations of regular and linear languages in terms of branching languages are obtained. Finally, the relations between branching grammars and contextual grammars of [5] are investigated.

1. INTRODUCTION

In the last time, in the formal languages theory many generative devices different from the Chomsky grammars were considered. The present paper deals with two known such devices from "unusual automata theory", namely, *finite branching automata* (shortly, *FBA*) of Havel [2] and *contextual grammars* of Marcus [5]. The *FBA*'s recognize families of languages by specifying how their strings branch to the right. The latter devices generate languages starting from a finite set of strings and adjoining contexts selected by means of a choice mapping.

The branching grammars incorporate features of both the two devices: they define languages by branching the strings in dependence on their suffixes and prefixes of a bounded length.

Two types of branching grammars (shortly, *BG*) are considered: *simple BG*'s (*SBG*) and *double BG*'s (*DBG*). In the former kind of *BG*'s only prolongations to the right are possible whereas in the latter, the words may be prolonged in both sides.

The two types of *BG*'s introduce infinite hierarchies of regular and, respectively, linear languages. Two interesting results about these generative devices are Theorems 2 and 6 giving homomorphic characterisations of regular and linear languages in terms of simple and double branching languages, respectively. The connection between *DBG*'s and contextual grammars (simple and with choice) is investigated, as well as the closure properties of the considered families of languages.

Let V be a vocabulary. We denote by V^* the free monoid generated by V under the operation of concatenation and the null element λ. The length of $x \in V^*$ is denoted by $|x|$.

For further notions concerning the formal languages theory see [9]. We merely specify that we denote by $G = (V_N, V_T, S, P)$ a Chomsky grammar with the nonterminal vocabulary V_N, the terminal vocabulary V_T, the start symbol S and the set of production rules P. Also, we denote by $\mathcal{L}_i$, $i = 0, 1, 2, 3$, the four families of languages in Chomsky's hierarchy.

Let $L \subseteq V^*$. (We use $\subseteq$ for inclusion and $\subset$ for strict inclusion.) Following [2] we put

$$\operatorname{Pref} L = \{u \in V^* \mid \text{there is } v \in V^* \text{ such that } uv \in L\}.$$

For $u \in V^*$ we define

$$\partial_u L = \{v \in V^* \mid uv \in L\}.$$

Then, the branching structure of L is described by the mapping $\Delta_L : V^* \to \mathcal{P}(V_\lambda)$ (V_λ stands for $V \cup \{\lambda\}$) defined by

$$\Delta_L(u) = (\operatorname{Pref} \partial_u L \cap V) \cup (\partial_u L \cap \{\lambda\}).$$

The language L is completely identified by Δ_L.

Now, let us recall from [2] the definition of *FBA*'s:

$$\mathcal{A} = (V, Q, \delta, q_0, B)$$

where V is a vocabulary, Q is a set of states, $q_0 \in Q$ is the initial state, $\delta : Q \times V \to Q$ is the next-state function and $B \subseteq Q \times \mathcal{P}(V_\lambda)$ is the set of branches.

This automaton was intended to recognize families of languages and not single languages ([2], [3]). In what follows, we consider that $B \subseteq Q \times V_\lambda$ and thus the automaton naturally identifies one language.

In this aim we extend δ to $Q \times V^*$ in the usual way and say that a string $x = x_1 \ldots x_n$, $x_i \in V$ is accepted by $\mathcal{A}$ if and only if there exist $q_0, q_1, \ldots, q_n$ in Q such that $\delta(q_{i-1}, x_i) = q_i$ and $(q_i, x_{i+1}) \in B$ for any i and $(q_n, \lambda) \in B$. (The null string is accepted only if $(q_0, \lambda) \in B$.)

Two components of this machinery (which is, in fact, a finite automaton with a branching controller) co-operate in selecting the strings of the recognized language: the mapping δ and the branch set B.

It is easy to see that a language is recognized by a *FBA* as above if and only if it is a regular language. Thus, we have two possibilities to go further: either to renounce to some conditions in the *FBA* definition or to impose additional restrictions in order to recognize a larger family of languages. In this paper we follow the first alternative.

Thus, let us consider that the mapping δ depends only on its second argument, that is, $\delta(q, a) = \delta(q', a)$, for any $q, q' \in Q$, $a \in V$. Then the branches depend only on symbols in V. The recognized language is determined by q_0, the allowed branches and the "final" states of Q (states for which a pair (q, λ) is in B). We completely eliminate the states but we extend the dependence of the branches to more symbols.

Definition 1. A *simple branching grammar of degree k* (a *k-SBG*) is a system

$$\mathscr{A} = (V, L_0, B),$$

where V is a vocabulary, $L_0 \subseteq V_0^k$ and $B \subseteq V_1^k \times V_\lambda$ (where $V_i^j = \{x \in V^* \mid i \leqq |x| \leqq j\}$).

Two languages generated by this grammar are defined in the following way.

For two languages L_1, L_2 on an arbitrary vocabulary V let $D(L_1, L_2)$ be the smallest language $L \subseteq V^*$ which includes L_1 and has the following property: if $x \in L$, $x = x_1x_2$ and $x_2a \in L_2$ for some $x_1, x_2 \in V^*$, $a \in V$, then $xa \in L$.

Then, for a given *SBG*, $\mathscr{A}$, as above, the *weakly generated language* is $D(L_0, L_B)$, where $L_B = \{xa \mid (x, a) \in B\}$. Denote it by $W(\mathscr{A})$.

The *strongly generated language* is

$$L(\mathscr{A}) = W(\mathscr{A}) \cap (\{\lambda\} \cup \{x_1x_2 \mid x_1 \in V^*, (x_2, \lambda) \in B\}).$$

We denote by $\mathscr{S}_k$ the family of strongly generated languages by *k-SBG*'s. Obviously, $\mathscr{S}_k \subseteq \mathscr{S}_{k+1}$. We define then

$$\mathscr{S}^\infty = \bigcup_{i=1}^{\infty} \mathscr{S}_i .$$

Theorem 1. We have $\mathscr{S}_1 \subset \mathscr{S}_2 \subset \ldots \subset \mathscr{S}^\infty \subset \mathscr{L}_3$.

Proof. In [7] it was proved that for any regular languages L_1, L_2, the language $D(L_1, L_2)$ is regular. As $\mathscr{L}_3$ is closed under intersection, it follows that $L(\mathscr{A})$ is regular for any $\mathscr{A}$.

To prove that $\mathscr{S}_{k-1} \subseteq \mathscr{S}_k$ is a proper inclusion, let us consider the language $L_k = \{a^k\}$. Obviously, $L_k = L(\mathscr{A})$ for $\mathscr{A} = (\{a\}, \{a^k\}, \{(a, \lambda)\})$. Therefore, $L_k \in \mathscr{S}_k$. Let us suppose that $L_k \in \mathscr{S}_{k-1}$, $L_k = L(\mathscr{A}')$ for $\mathscr{A}' = (\{a\}, L_0, B)$. Any $x \in L_0$ has $|x| \leqq k - 1$ hence at least a pair (a^i, a) exists in B. A pair (a^j, λ) belongs to B too. It follows that $L(\mathscr{A}')$ is infinite. Contradiction.

Let us consider now the regular language

$$L = \{a^nba^mc \mid n, m \geqq 1\}$$

and suppose that $L = L(\mathscr{A})$ for $\mathscr{A} = (V, L_0, B)$, $B \subseteq V_1^k \times V_\lambda$. In L there are strings containing sequences of a of arbitrary length and therefore a pair (a^i, a) must belong to B. On the other hand, pairs (a^j, b) and (a^r, c) exist in B as well as a pair (a^tc, λ).

Consequently, in $L(\mathscr{A})$ there are strings which contain more than one symbol b. Such strings are not in Lhence $L \neq L(\mathscr{A})$. □

A homomorphism $h : V_1 \to V_2$ is called a coding. An interesting property of *SBG*'s is the following.

Theorem 2. A language L is regular if and only if there is a coding h and a language $L' \in \mathscr{S}_1$ such that $L = h(L')$.

Proof. Let $\mathscr{A} = (V, Q, \delta, q_0, F)$ be a deterministic finite automaton. We construct the following 1-*SBG*: $\mathscr{A}' = (V', L_0, B)$, where

$$\begin{aligned} V' &= \{[a, q] \mid a \in V,\ q \in Q\}\,, \\ L_0 &= \{[a, q] \mid \delta(q_0, a) = q,\ q \in Q,\ a \in V\}\,, \\ B &= \{([a, q], [a', q']) \mid \delta(q, a') = q',\ q, q' \in Q,\ a, a' \in V\} \cup \\ &\cup \{([a, q], \lambda) \mid q \in F,\ a \in V\}\,. \end{aligned}$$

Let $h : V' \to V$ be the coding defined by $h([a, q]) = a$, $a \in V$, $q \in Q$. We have $L(\mathscr{A}) = h(L(\mathscr{A}'))$.

Indeed, let $x \in L(\mathscr{A})$, $x = x_1 \dots x_n$, $x_i \in V$ for all i. There exist $q_0, q_1, \dots, q_n \in Q$ with $q_i = \delta(q_{i-1}, x_i)$, $i = 1, 2, \dots, n$, $q_n \in F$. Consequently, $[x_1, q_1] \in L_0$, $([x_i, q_i], [x_{i+1}, q_{i+1}]) \in B$, $i = 1, 2, \dots, n - 1$ and $([x_n, q_n], \lambda) \in B$. Hence, $[x_1, q_1] \dots \dots [x_n, q_n] \in L(\mathscr{A}')$ and $x_1 \dots x_n = h([x_1, q_1] \dots [x_n, q_n]) \in h(L(\mathscr{A}'))$.

Conversely, let $x = [x_1, q_1] \dots [x_n, q_n] \in L(\mathscr{A}')$. As $[x_1, q_1] \in L_0$, it follows that $\delta(q_0, x_1) = q_1$. As $([x_i, q_i], [x_{i+1}, q_{i+1}]) \in B$, it follows that $\delta(q_i, x_{i+1}) = q_{i+1}$, $i = 1, 2, \dots, n - 1$. Moreover, $([x_n, q_n], \lambda) \in B$ implies that $q_n \in F$. In consequence, $h(x) = x_1 \dots x_n \in L(\mathscr{A})$, hence $h(L(\mathscr{A}')) \subseteq L(\mathscr{A})$.

The other implication obviously follows from Theorem 1. (The family $\mathscr{L}_3$ is closed under arbitrary homomorphisms.) □

Definition 2 [9]. A family $\mathscr{L}$ of languages is called *AFL* iff it contains a nonempty language different from $\{\lambda\}$ and is closed under union, concatenation, $+$, λ-free homomorphisms, intersection with regular languages and inverse homomorphisms. A family $\mathscr{L}$ which is not closed under any of the previous operations is called *anti-AFL*.

Any *AFL* includes the family $\mathscr{L}_3$. According to Theorem 1, neither $\mathscr{S}_i$, nor $\mathscr{S}^\infty$ are *AFL*'s. In fact, we have,

Theorem 3. All the families $\mathscr{S}_i$, $i \geqq 2$ and $\mathscr{S}^\infty$ are *anti-AFL*'s.

Proof. 1) *Union.* Let us consider the languages

$$L_1 = \{a^n b \mid n \geqq 1\}^+,\ L_2 = \{a^n c \mid n \geqq 1\}^+\,.$$

Obviously, $L_1 = L(\mathcal{A}_1)$ where $\mathcal{A}_1 = (\{a, b\}, \{a\}, \{(a, a), (a, b), (b, a), (b, \lambda)\})$, therefore $L_1 \in \mathcal{S}_1$. Analogously, $L_2 \in \mathcal{S}_1$. The language $L_1 \cup L_2$ is not in $\mathcal{S}^\infty$. Let $\mathcal{A} = (\{a, b, c\}, L_0, B)$ be generating $L_1 \cup L_2$. The set B must contain a pair (a^i, a), a pair (a^j, b), a pair (a^r, c), one $(a^k b, a)$ and final pairs of the form $(a^s b, \lambda)$,$(a^t c, \lambda)$. By such branches we can obtain strings of the form $a^n b a^m c$. Such strings are not in $L_1 \cup L_2$, therefore $L_1 \cup L_2$ cannot be in $\mathcal{S}^\infty$.

2) *Concatenation.* Let

$$L_1 = \{a^n b \mid n \geqq 1\}\,, \quad L_2 = \{a^n c \mid n \geqq 1\}\,.$$

Obviously, $L_1, L_2 \in \mathcal{S}_1$. From the proof of Theorem 1 it follows that $L_1 L_2 \notin \mathcal{S}^\infty$.

3) *Iteration* $+$. Consider the language

$$L = \{ab^n a \mid n \geqq 0\}\,.$$

We have $L = L(\mathcal{A})$ with $\mathcal{A} = (\{a, b\}, \{aa, ab\}, \{(a, \lambda), (b, b), (b, a)\})$, therefore $L \in \mathcal{S}_2$. However, L^+ is not in $\mathcal{S}^\infty$. Indeed, let $\mathcal{A} = (\{a, b\}, L_0, B)$ be generating L^+. Since $aa \in L^+$, there is in B either a pair (a, λ) or a pair (aa, λ). On the other hand, there are in L^+ strings of the form $ab^n aab^m a$. Consequently, there are in $W(\mathcal{A})$ strings of the form $ab^n aa$. As either (a, λ) or (aa, λ) is in B, it follows that $ab^n aa$ is in $L(\mathcal{A})$ too. Contradiction.

4) *Homomorphisms.* In view of Theorem 2, the families $\mathcal{S}_i$, $i \geqq 2$, $\mathcal{S}^\infty$ are not closed under λ-free homomorphisms.

5) *Intersection with regular languages.* The language V^* is in $\mathcal{S}_1$ for any finite vocabulary V. For any $L \in \mathcal{L}_3 - \mathcal{S}^\infty$ we have then $L \cap V^* \notin \mathcal{S}^\infty$.

6) *Inverse homomorphisms.* Let $L = \{bc\}$ and consider the homomorphism $h : \{a, b, c\} \to \{b, c\}^*$ defined by $h(a) = \lambda$, $h(b) = b$, $h(c) = c$. Then

$$h^{-1}(L) = \{a^n b a^m c a^p \mid n, m, p \geqq 0\}\,.$$

Obviously, $L \in \mathcal{S}_1$ but $h^{-1}(L)$ is not in $\mathcal{S}^\infty$. The proof of the last assertion is similar to that used when we showed that $\{a^n b a^m c \mid n, m \geqq 1\}$ is not in $\mathcal{S}^\infty$. □

Remark. The family $\mathcal{S}_1$ is not an *anti-AFL* since it is closed under $+$. Indeed, let $\mathcal{A} = (V, L_0, B)$ with $L_0 \subseteq V_\lambda$, $B \subseteq V \times V_\lambda$. We construct the 1-*SBG*, $\mathcal{A}' = (V, L_0, B')$, with $B' = B \cup \{(a, b) \mid (a, \lambda) \in B,\ b \in L_0\}$. The inclusion $L(\mathcal{A})^+ \subseteq L(\mathcal{A}')$ is obvious. Conversely, let $x = x_1 \dots x_n \in L(\mathcal{A}')$, $x_1 \in L_0$, $(x_i, x_{i+1}) \in B$, $i = 1, 2, \dots, r - 1$, $(x_r, x_{r+1}) \notin B$ for the smallest r. It follows that $(x_r, x_{r+1}) \in B' - B$, hence $(x_r, \lambda) \in B$ and $x_{r+1} \in L_0$. Consequently, $x_1 \dots x_r \in L(\mathcal{A})$ and $y = x_{r+1} \dots x_n \in L(\mathcal{A}')$. By the iteration of this procedure, a decomposition $x = y_1 y_2 \dots y_k$, with $y_i \in L(\mathcal{A})$ can be obtained, therefore $L(\mathcal{A}') \subseteq L(\mathcal{A})^+$.

In *SBG*'s the strings can be prolonged only to the right. In what follows we consider devices which allow prolongations to the right as well as to the left.

Definition 3. A *double branching grammar of degree k* (a *k-DBG*) is a system

$$\mathscr{A} = (V, L_0, B),$$

where V is a vocabulary, $L_0 \subseteq V_0^k$ and $B \subseteq (V_\lambda \times V_1^k) \times (V_1^k \times V_\lambda)$.

The *weakly generated language*, denoted $W(\mathscr{A})$, is the smallest language $L \subseteq V^*$ for which

i) $L_0 \subseteq L$,

ii) if $x \in L$ and there are w, x_1, x_2, w' in V^* such that $x = wx_1 = x_2w'$ and $((\alpha, w), (w', \beta)) \in B$ for some $\alpha, \beta \in V_\lambda$, then $\alpha x \beta \in L$.

The *strongly generated language* is $L(\mathscr{A}) = W(\mathscr{A}) \cap (\{\lambda\} \cup \{x \in V^* |$ there are x_1, x_2, w, w' in V^* such that $x = wx_1 = x_2w'$ and $((\lambda, w), (w', \lambda)) \in B\})$.

We denote by $\mathscr{D}_k$ the family of languages strongly generated by *k-DBG*'s. Obviously, $\mathscr{D}_i \subseteq \mathscr{D}_{i+1}$. We define

$$\mathscr{D}^\infty = \bigcup_{i=1}^{\infty} \mathscr{D}_i\,.$$

Remark. The family $\mathscr{D}_1$ contains non-regular languages. Indeed, let us consider the *1-DBG*

$$\mathscr{A} = (\{a, b\}, \{a\}, \{((a, a), (a, b)), ((a, a), (b, b)), ((\lambda, a), (b, \lambda))\})\,.$$

It is easy to see that $L(\mathscr{A}) = \{a^n b^{n-1} \mid n \geqq 2\}$ and this language is not a regular one.

Theorem 4. We have $\mathscr{S}_i \subset \mathscr{D}_i \subset \mathscr{D}_{i+1}$ for any $i \geqq 1$.

Proof. Let $\mathscr{A} = (V, L_0, B)$ be a *k-SBG*. We construct the *k-DBG* $\mathscr{A}' = (V, L_0, B')$, where $B' = \{((\lambda, a), (x, \alpha)) \mid a \in V, (x, \alpha) \in B\}$.

Obviously, $L(\mathscr{A}) = L(\mathscr{A}')$.

According to the inclusions $\mathscr{D}_i \subseteq \mathscr{D}_{i+1}$, the above Remark and Theorem 1, it follows that $\mathscr{S}_i \subset \mathscr{D}_i$.

Now, let us consider again the language $L_k = \{a^k\}$.

Clearly, $L_k \in \mathscr{D}_k$. Let us suppose that $L_k \in \mathscr{D}_{k-1}$, $L_k = L(\mathscr{A})$ for some $\mathscr{A} = (\{a\}, L_0, B)$. Any $x \in L_0$ is of the form a^i with $i \leqq k - 1$. In B there is at least a pair $((\alpha, a^i), (a^j, \beta))$ with $\alpha, \beta \in \{a, \lambda\}$, $\alpha\beta \neq \lambda$. It follows that $W(\mathscr{A})$ is infinite. Because there is in B a pair $((\lambda, a^i), (a^j, \lambda))$, it follows that $L(\mathscr{A})$ is infinite too. Contradiction. □

Let $\mathscr{L}_{lin}$ be the family of linear languages.

Theorem 5. $\mathscr{D}^{\infty} \subset \mathscr{L}_{lin}$.

Proof. Let $L \in \mathscr{D}_k$, $L = L(\mathscr{A})$, for $\mathscr{A} = (V, L_0, B)$. We construct the following linear grammar $G = (V_N, V, S, P)$, where

$$V_N = \{S\} \cup \{[w, w'] \mid w,\ w' \in V^*,\ |w| = |w'| = k\}\,,$$

$P = \{S \to w \mid w \in L,\ |w| \leqq 2k + 1\} \cup \{S \to [z, z'] |$ there is $((\lambda, x), (x', \lambda))$ in B such that $z = xy$ and $z' = y'x'$ for some $y, y' \in V^*\} \cup \{[w, w'] \to \alpha[z, z']\,\alpha' |$ there are y, y' in V_λ such that $wy = \alpha z$, $y'w' = z'\alpha'$, $\alpha, \alpha' \in V_\lambda$ and there is $((\alpha, x), (x', \alpha'))$ in B such that $z = xu$ and $z' = vx'$ for some $u, v \in V^*\} \cup \{[w, w'] \to wxw' \mid x \in V_\lambda,\ wxw' \in W(\mathscr{A})\}$.

Let us firstly observe that if a string in $W(\mathscr{A})$ can be obtained from another the lengths of the two strings differ by 1 or 2. As L_0 contains only strings with $|x| \leqq k$, it follows that any string in $L(\mathscr{A})$ has in its derivations a string $y \in W(\mathscr{A})$ of length $2k$ or $2k + 1$.

The equality $L(\mathscr{A}) = L(G)$ holds.

Let $x \in L(\mathscr{A})$. If $|x| \leqq 2k + 1$ we have obviously $x \in L(G)$. For x with $|x| > 2k + 1$, let us suppose that $x = x_r \dots x_1 y_1 \dots y_{2k+1} x'_1 \dots x'_r$ with $x_i, x'_i, y_i \in V_\lambda$ such that $y_1 \dots y_{2k+1} \in W(\mathscr{A})$ and for each $i \geqq 1$ we have $((x_i, u_{i-1}), (u'_{i-1}, x'_i)) \in B$ and $u_{i-1} w u'_{i-1} = x_{i-1} \dots x_1 y_1 \dots y_{2k+1} x'_1 \dots x'_{i-1}$ for some $w \in V^*$.

Moreover, $((\lambda, u_r), (u'_r, \lambda)) \in B$ and $u_r w u'_r = x$ for some $w \in V^*$. Consequently, in P there are the rules 1) $S \to [u_r v_r, v'_r u'_r]$ with v_r and v'_r such that $|u_r v_r| = |v'_r u'_r| = k$, 2) $[u_i v_i, v'_i u'_i] \to x_i[u_{i-1} v_{i-1}, v'_{i-1} u'_{i-1}]\, x'_i$ with $|u_i v_i| = |v'_i u'_i| = k$ for $i = 1, 2, \dots, r$, 3) $[u_0 v_0, v'_0 u'_0] \to y_1 \dots y_{2k+1}$.

Using these rules, we can obtain a derivation of x in the grammar G, therefore $L(\mathscr{A}) \subseteq L(G)$.

Conversely, let $x \in L(G)$. If $|x| \leqq 2k + 1$, then x can be derived directly from S hence $x \in L(\mathscr{A})$. If $|x| > 2k + 1$, then $x = x_r \dots x_1 y_1 \dots y_{2k+1} x'_1 \dots x'_r$, $x_i, x'_i, y_i \in V_\lambda$ and there is a derivation of the form

$$S \Rightarrow [w_r, w'_r] \Rightarrow x_r[w_{r-1}, w'_{r-1}]\, x'_r \Rightarrow \dots$$
$$\dots \Rightarrow x_r \dots x_1[w_0, w'_0]\, x'_1 \dots x'_r \Rightarrow x_r \dots x_1 y_1 \dots y_{2k+1} x'_1 \dots x'_r\,.$$

From the definition of G it follows that there are u_i, u'_i and v_i, v'_i such that $w_i = u_i v_i$, $w'_i = v'_i u'_i$ and $((x_i u_i), (u'_i, x'_i)) \in B$, $i = 1, 2, \dots, r$. In addition, $((\lambda, u_r), (u'_r, \lambda)) \in B$ and $y_1 \dots y_{2k+1} \in W(\mathscr{A})$. It follows that $x \in L(\mathscr{A})$ hence $L(G) \subseteq L(\mathscr{A})$ and the equality $L(\mathscr{A}) = L(G)$ is proved.

Let us now consider the regular language $L = \{a^{3n} \mid n \geqq 1\}$.

This language is not in $\mathscr{D}^{\infty}$ as follows from the following Lemma.

Lemma 1. For any $L \subseteq \{a\}^*$, $L \in \mathscr{D}^\infty$, there is a positive integer p such that for any $x \in L$, $|x| > p$, there is $x' \in L$ such that $|x'| - |x| \leqq 2$.

Proof. Let $\mathscr{A} = (\{a\}, L_0, B)$ be a k-DBG and let

$$p_1 = \min \{\max (|w|, |w'|) \mid ((\lambda, w), (w', \lambda)) \in B\},$$

$$p_2 = \max \{|x| \mid x \in L_0\}.$$

Then we take $p = \max \{p_1, p_2\}$.

Indeed, if there is x in $L(\mathscr{A})$ with $|x| > p$, it follows that $L(\mathscr{A})$ is infinite. Moreover, any string in $W(\mathscr{A})$ of length greater than or equal to p_1 is in $L(\mathscr{A})$. As the lengths of two strings in $W(\mathscr{A})$ which can be obtained one from another, differ by 1 or 2, the lemma follows. □

An interesting result about DBG's, corresponding to Theorem 2 for SBG's is the following

Theorem 6. A language L is linear if and only if there is $L' \in \mathscr{D}_1$ and a homomorphism h such that $L = h(L')$.

Proof. From Theorem 5 we have $\mathscr{D}_1 \subset \mathscr{L}_{lin}$. As $\mathscr{L}_{lin}$ is closed under homomorphisms, one implication holds.

Conversely, let $L \in \mathscr{L}_{lin}$, $L = L(G)$ for a given λ-free grammar $G = (V_N, V, S, P)$. (If there is a rule $A \to \lambda$ in P then $A = S$ and S does not occur in the right side of any rule.)

There exists an obvious procedure transforming the grammar G into an equivalent linear grammar G' whose rules are of the form

$$A \to a, \quad a \in V,$$

$$S \to \lambda,$$

$$A \to \alpha B \beta \quad \text{for} \quad \alpha, \beta \in V_\lambda.$$

Assume hence that G has only rules of these forms.

We construct the DBG $\mathscr{A} = (V', L_0, B)$, where

$$V' = \{[\alpha, A] \mid \alpha \in V_\lambda, A \in V_N \cup \{T\}\},$$

$$L_0 = \{[\alpha, T] \mid A \to \alpha \text{ is in } P\} \cup \{\lambda \mid S \to \lambda \in P\},$$

$$\begin{aligned} B = \{&(([\alpha_1, A_1], [\alpha_2, A_2]), ([\alpha_3, A_2], [\alpha_4, A_1])) \mid \alpha_1, \alpha_2, \alpha_3, \alpha_4 \in V_\lambda \text{ and} \\ &A_1 \to \alpha_2 A_2 \alpha_3 \text{ is in } P\} \cup \{((\lambda, [\alpha_2, A_2]), ([\alpha_3, A_2], \lambda))| \text{ for } \alpha_2, \alpha_3 \in V_\lambda \text{ and} \\ &S \to \alpha_2 A_2 \alpha_3 \text{ is in } P\} \cup \{(([\alpha_1, A_1], [\alpha_2, T]), ([\alpha_2, T], [\alpha_4, A_1])) \mid \alpha_1, \alpha_2, \alpha_4 \\ &\text{in } V_\lambda \text{ and } A_1 \to \alpha_2 \text{ in } P\}. \end{aligned}$$

Then $L(G) = h(L(\mathscr{A}))$ for the homomorphism $h : V' \to V$ defined by $h([\alpha, A]) = \alpha$, $\alpha \in V_\lambda$, $A \in V_N \cup \{T\}$.

Indeed, let $x \in L$, $x = x_1 \dots x_n z y_n \dots y_1$ with $x_i, y_i, z \in V_\lambda$ be such that there is the derivation $S \Rightarrow x_1 A_1 y_1 \Rightarrow x_1 x_2 A_2 y_2 y_1 \Rightarrow \dots \Rightarrow x_1 \dots x_n A_n y_n \dots y_1 \Rightarrow x$. Then, we have $[z, T] \in L_0$ and $(([\alpha_i, A_{i-1}], [x_i, A_i]), ([y_i, A_i], [\beta_i, A_{i-1}])) \in B$, $\alpha_i, \beta_i \in V_\lambda$, $i = 1, 2, \dots, n$, $((\lambda, [x_1, A_1]), ([y_1, A_1], \lambda)) \in B$ and $(([\alpha_{n+1}, A_n], [z, T]), ([z, T], [\beta_{n+1}, A_n])) \in B$, $\alpha_{n+1}, \beta_{n+1} \in V_\lambda$. For $\alpha_i = x_{i-1}$ and $\beta_i = y_{i-1}$ we obtain a derivation in $\mathscr{A}$ for the string $w = [x_1, A_1] [x_2, A_2] \dots [x_n, A_n] [z, T] [y_n, A_n] \dots [y_1, A_1]$, hence $w \in L(\mathscr{A})$. Obviously, $x = h(w)$ thus $L \subseteq h(L(\mathscr{A}))$.

Conversely, let $w \in L(\mathscr{A})$, $w = [x_1, A_1] \dots [x_n, A_n]\ [z, T] [y_n, A_n] \dots [y_1, A_1]$ with $x_i, y_i, z \in V_\lambda$ be obtained using $[z, T] \in L_0$, $((\lambda, [x_1, A_1]), ([y_1, A_1], \lambda)) \in B$, $(([x_{i-1}, A_{i-1}], [x_i, A_i]), ([y_i, A_i], [x_{i-1}, A_{i-1}])) \in B$ for $i = 1, 2, \dots, n$ and $(([x_n, A_n], [z, T]), ([z, T], [y_n, A_n])) \in B$.

From the definition of $\mathscr{A}$ it follows that there are in P the rules $S \to x_1 A_1 y_1$, $A_{i-1} \to x_i A_i y_i$, $i = 2, \dots, n$ and $A_n \to z$. Using these rules we can obtain the derivation $S \Rightarrow x_1 \dots x_n z y_n \dots y_1$ in G. Since $h(w) = x$ it follows that $h(w) \in L(G')$, hence $h(L(\mathscr{A})) \subseteq L(G)$ and the equality is proved. □

From Theorems 5 and 6 it follows that $\mathscr{D}_i$ and $\mathscr{D}^\infty$ are not closed under homomorphisms. From Lemma 1 it follows that there are regular languages that are not in $\mathscr{D}^\infty$. Consequently, $\mathscr{D}_i$ and $\mathscr{D}^\infty$ are not closed under intersection with regular languages.

Open problem. Are the families $\mathscr{D}_i$ and $\mathscr{D}^\infty$ *anti-AFL*'s?

4. BRANCHING GRAMMARS AND CONTEXTUAL GRAMMARS

There is a strong connection between *DBG*'s and contextual grammars defined in [5].

Definition 4. [5] A *simple contextual grammar* (shortly, *SCG*) is a triple $G = (V, L_0, C)$, where V is a vocabulary, L_0 is a finite language on V and C is a finite set of contexts on V (pairs $\langle u, v \rangle$ with $u, v \in V^*$). The language generated by G is the smallest language $L' \subseteq V^*$ for which

i) $L_0 \subseteq L'$,

ii) if $x \in L'$ and $\langle u, v \rangle \in C$, then $uxv \in L'$.

Definition 5. [5] A *contextual grammar with choice* (shortly, *CCG*) is a system $G = (V, L_0, C, \varphi)$ where V, L_0, C are as above and φ is a mapping $\varphi : V^* \to \mathscr{P}(C)$. The language generated by G is the smallest $L' \subseteq V^*$ for which

i) $L_0 \subseteq L'$,

ii) if $x \in L'$ and $\langle u, v \rangle \in \varphi(x)$, then $uxv \in L'$.

Let us denote by $\mathscr{C}_S$ and $\mathscr{C}_C$ the two families of contextual languages.

According to the above definitions, double branching grammars can be viewed as contextual grammars with choice, the choice depending on the leftmost and rightmost subwords of length k. However, there are essential differences between the two "unusual" generative devices: in *DBG*'s the end of derivation is controlled, whereas this is not the case in *CCG*'s; on the other hand, the choice in *CCG*'s by means of φ is a stronger one.

Theorem 7. The family $\mathscr{D}^\infty$ and any family in $\{\mathscr{C}_S, \mathscr{C}_C\}$ are incomparable.

Proof. The following results about contextual languages were proved in [6]: $\mathscr{C}_S \subset \mathscr{C}_C$, $\mathscr{C}_S \subset \mathscr{L}_{lin}$, $\mathscr{L}_3 - \mathscr{C}_C \neq \emptyset$. Moreover, $\mathscr{C}_C$ and $\mathscr{C}_S$ are closed under homomorphisms [6]. If $\mathscr{D}^\infty \subseteq \mathscr{C}_C$ then, from Theorem 6, it would follow that $\mathscr{L}_{lin} \subseteq \mathscr{C}_C$. Contradiction.

In [6] the following necessary condition for a language to be contextual with choice was given.

For $x, y \in V^*$ let $x < y$ iff $y = uxv$. If $L \subseteq V^*$ we define

$$K^1(L) = \{x \in L \mid \text{there is no } y \in L \text{ such that } y < x\},$$

$$K^{i+1}(L) = K^1(L - K^i(L)).$$

For any $L \in \mathscr{C}_C$ and for any $i \geqq 1$ the set $K^i(L)$ is finite [6].

Now, consider the language $L = \{ab^n a \mid n \geqq 1\}$. Obviously, $K^1(L) = L$, therefore this language is not in $\mathscr{C}_C$. However, the

$$DBG\ \mathscr{A} = (\{a, b\}, \{b\}, \{((b, b),(b, \lambda)), ((a, b), (b, a)), ((\lambda, a), (a, \lambda))\})$$

obviously generates L, hence $L \in \mathscr{D}_1$.

In view of the inclusion $\mathscr{C}_S \subset \mathscr{C}_C$, the theorem is completely proved. □

Let $\mathscr{D}_w$ be the family of weakly generated languages by *DBG*'s.

Theorem 8. 1) The families $\mathscr{D}_w$ and $\mathscr{C}_S$ are incomparable. 2) $\mathscr{D}_w \subset \mathscr{C}_C$.

Proof. 1) Clearly, the language

$$L = \{a^n b^m a^m b^n \mid m \geqq 1,\ n \geqq 0\}$$

is in $\mathscr{D}_w$. It is easy to see that $\mathrm{Var}(L) = 2$. (Following [1], $\mathrm{Var}(G) = \mathrm{card}\, V_N$ for $G = (V_N, V_T, S, P)$, and $\mathrm{Var}(L) = \min\{\mathrm{Var}(G) \mid L = L(G)\}$.) Following [5], for any $L \in \mathscr{C}_S$, $\mathrm{Var}(L) = 1$. Consequently, the above language is not in $\mathscr{C}_S$.

On the other hand, Lemma 1 is true also for $\mathscr{D}_w$. Thus the language $L = \{a^{3n} \mid n \geqq 1\}$ is in $\mathscr{C}_S$ but not in $\mathscr{D}_w$.

2) Let $\mathscr{A} = (V, L_0, B)$. We construct the following *CCG*, $G = (V, L_0, C, \varphi)$, where

$$C = \{\langle\alpha, \beta\rangle \mid ((\alpha, w), (w', \beta)) \in B\}$$

and $\varphi : V^* \to \mathscr{P}(C)$ is defined by

$$\varphi(x) = \{\langle\alpha, \beta\rangle \mid \text{there are } w, x_1, x_2, w' \text{ in } V^* \text{ such that}$$
$$x = wx_1 = x_2w' \text{ and } ((\alpha, w), (w', \beta)) \in B\}\,.$$

It is easy to see that $W(\mathscr{A}) = L(G)$. □

An intermediate family between $\mathscr{C}_S$ and $\mathscr{C}_C$ was considered in [8]: the programmed contextual languages (shortly, *PCL*).

Definition 6. [8] A *programmed contextual grammar* is a system $G = (V, L_0, C, \varphi)$, where V, L_0, C are as above and φ is a mapping $\varphi : L_0 \cup C \to \mathscr{P}(C)$. The generated language is

$$L(G) = L_0 \cup \{u_n \ldots u_1xv_1 \ldots v_n \mid n \geqq 1,\ x \in L_0,\ \langle u_1, v_1\rangle \in \varphi(x)\,,$$
$$\langle u_i, v_i\rangle \in \varphi(\langle u_{i-1}, v_{i-1}\rangle)\,,\quad i = 2, 3, \ldots, n\}\,.$$

Open problem. Does the family of *PCL*'s include the family $\mathscr{D}_w$?

The converse is not true according to Theorem 8 and the inclusion of $\mathscr{C}_S$ in the family of *PCL*'s ([8]).

Note: Many useful remarks of the referee, allowing us to improve the form of this paper, are acknowledged.

(Received February 18, 1977.)

REFERENCES

[1] J. Gruska: Descriptional complexity of context-free languages. Proc. of Symp. and Summer School Math. Found. of Computer Sci., High Tatras 1973.

[2] I. M. Havel: Finite branching automata. Kybernetika *10* (1974), 281—302.

[3] I. M. Havel: On the branching structure of languages. Proc. of Symp. Math. Found. of Computer Sci., Gdansk 1976, Lecture Notes in Computer Science 45 (1976).

[4] S. Marcus: Gramatici şi automate finite. Ed. Academiei R.S.R., Bucureşti 1964.

[5] S. Marcus: Contextual grammars. Rev. Roum. Math. Pures et Appl. *10* (1969), 1525—1534.

[6] Gh. Păun: Asupra gramaticilor contextuale. Studii şi cercetări matematice *26* (1974), 1111—1129.

[7] Gh. Păun: On a prolongation operation of languages. Bull. Math. de la Soc. de Sci. Math. de R.S.R. (in press).

[8] Gh. Păun: Contextual grammars with restrictions in derivation. Rev. Roum. Math. Pures et Appl. *22* (1977), 1147—1154.

[9] A. Salomaa: Formal languages. Academic Press, New York — London 1973.

Dr. Gheorghe Păun, University of Bucharest, Division of Systems Studies, Str. Academiei 14, s. 1 Bucureşti. Romania 70109.