On some extensions of indian parallel context free grammars

By J. DASSOW

1. Introduction

In order to get better models for aspects of programming and natural languages some extensions of context free grammars are introduced, for instance matrix grammars, random context grammars, programmed grammars, time-variant grammars (see [1], [13], [7], [10], [11]), which are characterized by mechanisms regulating the use of the productions. The relations between the associated language families are studied by some authors (see [10], [4], [11]).

In [12], R. SIROMONEY and K. KRITHIVASAN regard a parallel version of context free grammars. In this paper we introduce some of the above mentioned extensions for these indian parallel context free grammars. An other generalization of the indian parallel grammars are the EDTOL systems (see [3]). We shall prove that all these language families coincide.

The result has also another interesting aspect. The EDTOL systems work purely parallel, i.e. all occurrences of all letters are rewritten in a single derivation step; the other extensions of the indian parallel grammar have a sequential aspect because only all occurrences of one letter are rewritten in a single step. Therefore our result can be regarded as a sequential characterization of EDTOL languages. Thus, it is of interest in connection with the sequential characterizations of ETOL languages (see [14], [6], [9], [8], [2]).

2. Definitions and notations

At first we recall the definition of the indian parallel context free grammar and its derivation process.

Indian context free grammar. An indian context free grammar is a construct $G=(V_N, V_T, P, S)$ where

i) V_N and V_T are finite nonempty sets, $V_N \cap V_T = \emptyset$,

ii) P is a finite subset of $V_N \times (V_N \cup V_T)^*$ (the elements of P are written as $A \to w$, $A \in V_N$, $w \in (V_N \cup V_T)^*$),

iii) $S \in V_N$.

Let $V=V_N\cup V_T$. Let $x\in V^+$, $y\in V^*$. We say that x directly derives y iff

i) $x=x_1Ax_2Ax_3\dots x_{n-1}Ax_n$, $A\in V_N$, $x_i\in(V\setminus\{A\})^*$,

ii) $y=x_1wx_2wx_3\dots x_{n-1}wx_n$,

iii) $A\to w\in P$.

Then we write $x\Rightarrow y$. Let $\overset{*}{\Rightarrow}$ be the reflexive and transitive closure of $\Rightarrow$.

The language $L(G)$ generated by G is defined as

$$L(G)=\{x\colon S\overset{*}{\Rightarrow}x,\ x\in V_T^*\}.$$

Now we define some extensions of this grammar by certain mechanisms regulating the derivation process. In all cases we use the alphabet V_N of nonterminals, the alphabet V_T of terminals, the axiom $S\in V_N$, productions $A\to w$, $A\in V_N$, $w\in V^*$, the application of a rule is in all cases defined as above, and if it is not stated otherwise then the associated language is defined in the way given above. We give the regulating mechanisms.

Indian matrix grammar. $G=(V_N, V_T, M, S)$ is an indian matrix grammar iff M is a finite set of finite sequences of productions,

$$M=\{m_1, m_2, \dots, m_r\},$$

$$m_i=[A_{i_1}\to w_{i_1}, A_{i_2}\to w_{i_2}, \dots, A_{i_s}\to w_{i_s}] \quad \text{for} \quad i=1,2,\dots,r.$$

The elements of M are called matrices. To apply such a matrix one has to apply the productions $A_{i_1}\to w_{i_1}, \dots, A_{i_s}\to w_{i_s}$ in the given order. Only those words of V_T^* are in $L(G)$ which are obtained by applications of the matrices.

Indian periodically time-variant grammars. An indian periodically time-variant grammar is a construct $G=(V_N, V_T, P, S, f)$, where f is a mapping $\mathbf{N}\to\mathfrak{P}(P)$ such that $f(i+j)=f(i)$ where m and j are fixed and $i>m$ is arbitrary. The derivation is regulated by the condition that the production used in the k-th step has to be in the set $f(k)$.

Indian random context grammar. The productions of an indian random context grammar $G=(V_N, V_T, P, S)$ are of the form

$$A\to w, R, Q$$

where R and Q are subsets of V_N. Such a production is only applicable on a word $x=x_1Ax_2Ax_3\dots x_{n-1}Ax_n$ if $x_1x_2\dots x_{n-1}x_n$ contains no letter of R and contains all letters of Q.

Indian programmed context free grammar. The productions of an indian programmed grammar $G=(V_N, V_T, P, S)$ are of the form

$$(l)\quad A\to w, F, S$$

where l is the label of the production, F and S are sets of labels. If $A\to w$ is applicable to x, then the next production has to be a rule with a label contained in the success field S. If $A\to w$ is not applicable then the next production has to have a label contained in the failure field F.

All these grammars work in a sequential-parallel way, i.e. only one letter is rewritten in a single derivation step, but all occurrences of this letter are rewritten. Starting from biological motivations EDTOL languages are defined which are also a generalization of indian parallel context free languages. The associated grammars work purely parallel as it is seen from the following definition.

EDTOL system. An EDTOL system is a construct

$$G=(V, V_T, \{P_1, P_2, \dots, P_r\}, S)$$

where

i) V is a finite set, V_T is a nonempty subset of V,
ii) $S\in V\setminus V_T$,
iii) each P_i is a finite subset of $V\times V^*$, the projection of P_i on the first coordinate is V, and if $A\to w_1$, $A\to w_2$ are in P_i then $w_1=w_2$.

(Usually only $S\in V^+$ is required. It is easy to prove that $S\in V\setminus V_T$ does not restrict the generative power.)

Let $x\in V^+$ and $y\in V^*$. It is said that x directly derives y (also written $x\Rightarrow y$) iff

i) $x=x_1x_2\dots x_n$, $x_i\in V$,
ii) $y=y_1y_2\dots y_n$,
iii) there is a $j\in\{1, 2, \dots, r\}$ such that $x_i\to y_i\in P_j$ for $i=1, 2, \dots, n$.

The language $L(G)$ is again defined as

$$L(G) = \{x\colon S\overset{*}{\Rightarrow} x, x\in V_T^*\}.$$

We use the following notations

$\mathscr{F}$ (IM) — family of indian matrix languages,
$\mathscr{F}$ (IPTV) — family of indian periodically time-variant languages,
$\mathscr{F}$ (IRC) — family of indian random context languages,
$\mathscr{F}$ (IPCF) — family of indian programmed context free languages,
$\mathscr{F}$ (EDTOL) — family of EDTOL languages.

3. Comparison of the language families

In the following proofs we will often introduce new alphabets. We make the next convention: If $U\subseteq V$ and $V_U'=\{x'\colon x\in U\}$ is a new alphabet then $w'=x_1''x_2''\dots x_n''$ for $w=x_1x_2\dots x_n$ where

$$x_i''=\begin{cases}x_i' & x_i\in U\\ x_i & x_i\notin U.\end{cases}$$

Let min (w) denote the set of letters occurring in w.

Lemma 1. $\mathscr{F}$(EDTOL)$\subseteq\mathscr{F}$(IM)

Proof. Let $G=(V, V_T, \{P_1, P_2, \dots, P_r\}, S)$ be an EDTOL system. Let $w_{x,P}$ be the right side of the production with the left side x in $P\in\{P_1, P_2, \dots, P_r\}$. Further we put $f(U, P)=\bigcup_{x\in U}\min(w_{x,P})$ for $U\subseteq V$.

For a subset $U\subseteq V$ we introduce a new alphabet $V_U=\{x_U\colon x\in U\}$. Now we define the following matrices for $U=\{x_{i_1}, x_{i_2}, \dots, x_{i_k}\}\subseteq V$ and $P\in\{P_1, P_2, \dots, P_r\}$,

$$P_U=[(x_{i_1})_U\to(w_{x_{i_1},P})_{f(U,P)}, (x_{i_2})_U\to(w_{x_{i_2},P})_{f(U,P)}, \dots, (x_{i_k})_U\to(w_{x_{i_k},P})_{f(U,P)}],$$

$$Q_U=[(x_{i_1})_U\to x_{i_1}, (x_{i_2})_U\to x_{i_2}, \dots, (x_{i_k})_U\to x_{i_k}],$$

and consider the indian matrix grammar

$$H = \left((V\setminus V_T)\cup \bigcup_{U\subseteq V} V_U, V_T, \bigcup_{i=1}^{r} \bigcup_{U\subseteq V} (P_i)_U \cup \bigcup_{U\subseteq V} Q_U, S_{\{S\}}\right).$$

The application of the matrix P_U models the application of P to words w with $U=\min(w)$, i.e. if $w\Rightarrow w'$ is in G then $w_U\Rightarrow(w')_{U'}$ is in H where $U=\min(w)$, $U'=\min(w')$. The application of the matrix Q_U is a translation of w_U in w. If $w\in V_U^*$ then we can apply only the matrices P_U and Q_U. Now it is easy to see that $L(G)=L(H)$. Therefore $L(G)\in\mathcal{F}(\mathrm{IM})$.

Lemma 2. $\mathcal{F}(\mathrm{IM})\subseteq\mathcal{F}(\mathrm{IRC})$.

Proof. Let $L\in\mathcal{F}(\mathrm{IM})$ and $L=L(G)$ for the indian matrix grammar $G=(V_N, V_T, M, S)$. Let $M=\{m_1, m_2, ..., m_r\}$, $m_i=[A_{i_1}\to w_{i_1}, A_{i_2}\to w_{i_2}, ..., ..., A_{i_s}\to w_{i_s}]$ for $i=1, 2, ..., r$. We introduce new alphabets $V^{i,j}=\{x^{i,j}\colon x\in V\}$, $1\leqq i\leqq r$, $1\leqq j\leqq s-1$. Let $V'=\bigcup_{i=1}^{r}\bigcup_{j=1}^{s-1} V^{i,j}\cup V_N$.

Now we can model the application of the matrix m_i by the following sequence of productions of an indian random context grammar

$$A_{i_1} \to (w_{i_1})^{i,1}, V'\setminus V_N, \emptyset$$
$$x \to x^{i,1}, V'\setminus(V_N\cup V^{i,1}), \min\left((w_{i_1})^{i,1}\right) \quad \text{for} \quad x\in V_N,$$
$$A_{i_2}^{i,1} \to (w_{i_2})^{i,2}, V\setminus V^{i,1}, \emptyset$$
$$x^{i,1} \to x^{i,2}, V\setminus(V^{i,1}\cup V^{i,2}), \min\left((w_{i_2})^{i,2}\right) \quad \text{for} \quad x^{i,1}\in V^{i,1}$$
$$\vdots$$
$$A_{i_s}^{i,s-1} \to (w_{i_s}), V\setminus V^{i,s-1}, \emptyset$$
$$x^{i,s-1} \to x, V\setminus(V^{i,s-1}\cup V_N), \min(w_{i_s}) \quad \text{for} \quad x^{i,s-1}\in V^{i,s-1}.$$

If we consider only such productions in our indian random context grammar then we can have also only such derivations which model the application of matrices. Therefore, we generate the same language. Thus $L\in\mathcal{F}(\mathrm{IRC})$.

The above construction works correctly only if we have no rule of form $A_{i_j}\to\lambda$ in the matrices (λ denotes the empty word). If we have a matrix m such that

$$m = [A_1 \to w_1, ..., A_j \to \lambda, ..., A_k \to w_k]$$

we use

$$m' = [A_1 \to w_1, ..., A_j \to B_j, ..., A_k \to w_k, B_j \to \lambda]$$

instead of m, where B_j is a new nonterminal. It is easy to see that this modification do not change the generative capacity and that our construction works also in the modified case.

Lemma 3. $\mathcal{F}(\mathrm{IRC})\subseteq\mathcal{F}(\mathrm{IPCF})$.

Proof. Let $G=(V_N, V_T, P, S)$, $V_N=\{A_1, A_2, ..., A_n\}$. We give a possibility to model a production of G by rules of an indian programmed context free grammar.

Let $A_j \to w$, $\{A_{i_1}, A_{i_2}, \ldots, A_{i_s}\}$, $\{A_{j_1}, A_{j_2}, \ldots, A_{j_t}\}$ be a rule of G. Consider a new alphabet $V' = \{x' : x \in V_N\}$ associated with the rule and the following diagram of IPCF-productions (an arc labelled by F connects a production with its failure field, and an arc labelled by S connects it with its success field).

It is obvious that we can only simulate rules of G because we have associated the primed alphabets with the productions of G. This proves $L(G) \in \mathscr{F}(\mathrm{IPCF})$.

Lemma 4. $\mathscr{F}(\mathrm{IPCF}) \subseteqq \mathscr{F}(\mathrm{EDTOL})$.

Proof. Let $L \in \mathscr{F}(\mathrm{IPCF})$, $L = L(G)$ for an indian programmed context free grammar $G = (V_N, V_T, P, S)$. Let R be the set of labels of the productions of P. With each subset I of R we associate new alphabets $V_I = \{x_I : x \in V_N\}$ and $V'_I = \{x'_I : x \in V_N\}$. We define tables $P_{l,I,i}$ for any $l \in I$, $I \subseteqq R$, $i \in \{1, 2, 3\}$ in the following way: If (l) $A \to w, F, S$ is a rule and $l \in I$ then put

$$P_{l,I,1} = \{x_I \to x_F : x \neq A\} \cup \{A_I \to f, f \to f, S \to S\} \cup \{x \to x : x \in V_T\},$$

$$P_{l,I,2} = \{x_I \to x_I : x \neq A\} \cup \{A_I \to A'_I, f \to f, S \to S\} \cup \{x \to x : x \in V_T\},$$

$$P_{l,I,3} = \{x_I \to x_S : x \neq A\} \cup \{A'_I \to w_S, f \to f, S \to S\} \cup \{x \to x : x \in V_T\}.$$

Using productions from sets of these types we can only generate words which contain only letters of V_T and V_I for a certain I with exception of one letter which can be in V'_I. If we have such a word we can apply only tables $P_{l,I,i}$. Further $P_{l,I,1}$ models the case that A_I does not occur and produces a word which consist of terminals and nonterminals of the alphabet associated with the failure field. The other two tables model the application of $A \to w$, and we get a word with nonterminals of the alphabet associated with the success field.

Let $I_1, I_2, \ldots, I_r$ be the sets containing labels whose production has the left side S, and put

$$Q_i = \{S \to S_{I_i}, f \to f\} \cup \{x \to x : x \in V_T \cup \bigcup_{I \subseteqq R} (V_I \cup V'_I)\}.$$

Then the EDTOL system

$$H = (\{S, f\} \cup V_T \cup \bigcup_{I \subseteqq R} (V_I \cup V'_I), V_T, \{P_{l,I,i} : I \subseteqq R, i \in \{1, 2, 3\}, l \in I\} \cup \cup \{Q_i : 1 \leqq i \leqq r\}, S)$$

generates L. Thus $L \in \mathscr{F}(\mathrm{EDTOL})$.

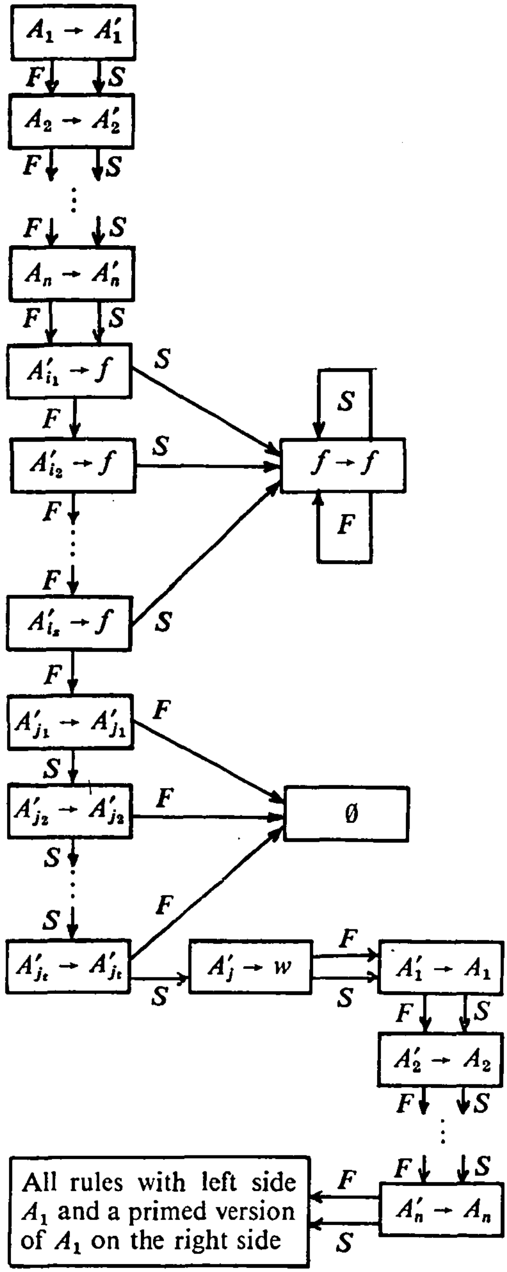
Lemma 5. $\mathscr{F}(\mathrm{IM}) \subseteqq \mathscr{F}(\mathrm{IPTV})$.

Proof. The proof of [10], Theorem 11 works also in the indian parallel case.

Lemma 6. $\mathscr{F}(\mathrm{IPTV}) \subseteqq \mathscr{F}(\mathrm{EDTOL})$.

Proof. Let $L = L(G)$ for the indian periodically time-variant grammar $G = (V_N, V_T, P, S, f)$ where $f(i+j) = f(i)$ for $i > m$. We introduce new alphabets $V^{(k)} = \{x^{(k)} : x \in V_N\}$ for $1 \leqq i \leqq m+j-1$. For $A \to w = p \in P$, $p \in f(i)$, $1 \leqq i < m+j-1$ we define the tables

$$P_{i,p} = \{x^{(i)} \to x^{(i+1)} : x \neq A\} \cup \{A^{(i)} \to w^{(i+1)}\} \cup \{x \to x : x \in V_T\}$$



$A_1 \to A_1'$
$A_2 \to A_2'$
$A_n \to A_n'$
$A_{i_1}' \to f$
$A_{i_2}' \to f$
$A_{i_s}' \to f$
$f \to f$
$A_{j_1}' \to A_{j_1}'$
$A_{j_2}' \to A_{j_2}'$
$A_{j_t}' \to A_{j_t}'$
$\emptyset$
$A_j' \to w$
$A_1' \to A_1$
$A_2' \to A_2$
$A_n' \to A_n$
All rules with left side A_1 and a primed version of A_1 on the right side
F
S

Fig. 1

and

$$P_{m+j-1,p} = \{x^{(m+j-1)} \to x^{(m)}: x \neq A\} \cup \{A^{(m+j-1)} \to w^{(m)}\} \cup \{x \to x: x \in V_T\}.$$

It is easy to see that the EDTOL system

$$H = \left(V_T \cup \bigcup_{k=1}^{m+j-1} V^{(k)}, V_T, \{P_{i,p}: 1 \leq i \leq m+j-1, p \in f(i)\}, S^{(1)}\right)$$

generates L.

We say that a grammar is λ-free if it contains no production with the empty word λ at the right side. The family of X-languages generated by λ-free X-grammars is denoted by $F_\lambda(X)$. As usual we identify languages which differ only in the empty word.

Theorem 1. $\mathscr{F}(\text{EDTOL}) = \mathscr{F}(\text{IM}) = \mathscr{F}(\text{IRC}) = \mathscr{F}(\text{IPCF}) = \mathscr{F}(\text{IPTV}) = \mathscr{F}_\lambda(\text{EDTOL}) = \mathscr{F}_\lambda(\text{IM}) = \mathscr{F}_\lambda(\text{IRC}) = \mathscr{F}_\lambda(\text{IPCF}) = \mathscr{F}_\lambda(\text{IPTV})$.

Proof. The first row follows directly by Lemma 1—6. Further $\mathscr{F}(\text{EDTOL}) = \mathscr{F}_\lambda(\text{EDTOL})$ is known and all our constructions in Lemma 1—4 and 6 preserve λ-freeness. If the matrix grammar in the proof of [10], Theorem 11 is λ-free then we modify the proof in the following way: The new symbols Y_j^i are not catenated with P_j^i, the last letter of P_j^i has to be in a new "primed" alphabet and the last rules have to change the letter into a "not primed" letter.

By Theorem 1, we get some information on properties of the extensions of indian parallel context free grammars, because we have knowledge on $\mathscr{F}(\text{EDTOL})$.

— In [3], closure properties under AFL-operations are given.

— There are context free languages which are not in $\mathscr{F}(\text{EDTOL})$.

— It is known that the families of matrix languages, programmed context free languages, random context languages and periodically time-variant languages properly contain $\mathscr{F}(\text{EDTOL})$. Thus the indian parallel restriction reduces the generative capacity of the considered extensions.

— The proof of v. Solms [14] works also in the indian parallel and deterministic case. This proves that indian random context grammars of special type generate already all indian random context languages.

A further language family which is equal to the above families is given in [2], Theorem 2.

Finally we want to mention without proof that all our language families also coincide with the family of indian unordered scattered context languages, which are the indian parallel version of the unordered scattered context grammars of [5].

TECHNOLOGICAL UNIVERSITY OTTO VON GUERICKE
DEPARTMENT OF MATHEMATICS AND PHYSICS
MAGDEBURG, GRD

References

[1] Abraham, S., Some questions of phrase structure grammars, *Comput. Linguistics*, v. 4, 1965, pp. 61—70.

[2] Dassow, J., ETOL systems and compound grammars, Rostock, Math. Colloq., v. 11, 1979, pp. 41—46.

[3] EHRENFEUCHT, A. & G. ROZENBERG, On images of inverse homomorphisms, *Automata, Languages, Development*, Ed. A. Lindenmayer, G. Rozenberg, North-Holland, 1976.
[4] MAYER, O., Some restrictive devices for context-free grammars, *Inform. and Control*, v. 20, 1972, pp. 69—92.
[5] MILGRAM, D. & A. ROZENFELD, A note on scattered context grammars, *Inform. Process. Lett.*, v. 1, 1971, pp. 47—50.
[6] PENTTONEN, M., ETOL-grammars and N-grammars, *Inform. Process. Lett.*, v. 4, 1975, pp. 11—13.
[7] ROSENKRANTZ, D. J., Programmed grammars and classes of formal languages, *J. Assoc. Comput. Mach.*, v. 10, 1969, pp. 107—131.
[8] ROZENBERG, G., More on ETOL systems versus random context grammars, *Inform. Process. Lett.*, v. 5, 1976, pp. 102—106.
[9] ROZENBERG, G. & D. VERMEIR, Context-free programmed grammars and ETOL systems, *Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science, v. 45, 1976, pp. 482—487.
[10] SALOMAA, A., On grammars with restricted use of productions, *Ann. Acad. Sci. Fenn. Ser. A I Math.*, No. 454, 1969.
[11] SALOMAA, A., *Formal languages*, Academic Press, 1973.
[12] SIROMONEY, R. & K. KRITHIVASAN, Parallel context-free languages, *Inform. and Control*, v. 24, 1974, pp. 155—162.
[13] VAN DER WALT, A. P. J., Random context languages, *Information Processing* 1971, North-Holland, 1972, pp. 68—66.
[14] VON SOLMS, S. H., On TOL languages over terminals, *Inform. Process. Lett.*, v. 3, 1975, pp. 69—70.

(Received June 6, 1979)