Peter Růžička
On the size of DeRemer's analyzers

Kybernetika, Vol. 11 (1975), No. 3, (207)--217

Persistent URL: http://dml.cz/dmlcz/124954

KYBERNETIKA — VOLUME 11 (1975), NUMBER 3

On the Size of DeRemer's Analyzers

PETER RUŽIČKA

The paper presents an upper estimation of the number of states of DeRemer's automaton $\mathscr{A}_G$ for a context - free grammar G. The estimation is given in terms of the number of elements of some sets constructed from the rules of G.

1. INTRODUCTION

LR(k) grammars form the largest known class of context-free (CF) grammars for which deterministic canonical nonbacktracking parsing algorithm can be mechanically constructed. Knuth [2] has proposed a method testing whether any CF grammar G for an arbitrary integer $k \geqq 0$ fulfils the LR(k) condition. It simultaneously produces the parsing algorithm for G. The size of the crucial part of this parsing algorithm called parsing table grows exponentially with the complexity of the grammar and therefore the mentioned algorithm is of no use in practical cases. A great effort has been made to optimize the number of states of LR(k) parsing table (one size of the parsing table is in fact represented by the number of states). One of the main results in this area is DeRemer's method constructing parsers for a special class of LR(k) grammars called SLR(k) grammars.

For any CF grammar G DeRemer shows the way how to construct a characteristic finite state automaton $\mathscr{A}_G$ (further denoted as DR automaton) accepting characteristic language of the grammar G. DR automaton is defined in section 3. If G is LR(0) grammar, then DR automaton $\mathscr{A}_G$ represents a parser for given grammar. If G is SLR(k) grammar for $k > 0$, then DR automaton forms the basis for the recognizer which in ambiguous situations has to make decisions based on look – ahead sets of words with the length up to k symbols.

In this paper the upper estimation of the number of states for DR automaton $\mathscr{A}_G$ is expressed in terms of productions of the grammar G. If G is an ε-free LL(1)

grammar, then this estimation is exact and it is identical with the estimation given in the paper by Král and Demner [3]. The quality of the estimation is further discussed. Another estimation expressed as a function of characteristics not depending on derivation relations is formulated.

2. NOTATION AND PRELIMINARY DEFINITIONS

In the paper we adopt the following notation and conventions. A context-free grammar G is designated by the quadruple $\langle V, \Sigma, P, S\rangle$ where V (alphabet) is a finite set of terminal and nonterminal symbols, Σ is a set of terminal symbols, P is a finite set of pairs called rules (productions) and S is distinguished starting symbol from $V - \Sigma$. Each rule is of the form $A \to \alpha$ where $A \in V - \Sigma$ and $\alpha \in V^*$ are left side and right side of the production respectively. V^* denotes the set of all finite strings formed from V including the empty string ε. Let production of the grammar be ordered in a sequence. The production Π will be written in the form $\Pi\colon A_\Pi \to \alpha_\Pi$. We denote the length of a string α by $|\alpha|$. If i is an integer, then $h(\alpha, i)$ is the prefix of α of the length i. $H(\alpha, i) = \{h(\alpha, j) \mid 1 \leqq j \leqq i\}$. Furthermore, $p(\mathscr{U})$ represents the number of elements of the finite set $\mathscr{U}$. The empty set is denoted by $\emptyset$ and for a nonterminal symbol A the set of 'left' symbols by $\mathscr{L}(A) = \{B \mid A \Rightarrow^* B\alpha, \alpha \in V^*, B \in V\}$. A nonterminal symbol A is called left recursive (infix), if $A \in \mathscr{L}(A)$ (if there exists a rule $A_\Pi \to \alpha_1 A \alpha_2$ in P such that $\alpha_1 \neq \varepsilon$).

3. DR AUTOMATON

Let us briefly review the construction of DeRemer's parsing algorithm. For our purposes a new production $S' \to \bot\, S \bot$ is added to the grammar G where S' becomes the new starting symbol of G and S', $\bot$ are distinct symbols not previously in V. The basic ingredients of DeRemer's parsing algorithm are configuration sets. Their members, called configurations, have the form $A_\Pi \to \alpha_{\Pi_1} . \alpha_{\Pi_2}$ where $A_\Pi \to \alpha_{\Pi_1}\alpha_{\Pi_2}$ is a production of G and the configuration marker indicates the position of parsing. A configuration is said to be the initial one if it has the form $A_\Pi \to . \alpha_\Pi$. A symbol situated immediately to the right of the configuration marker is called the successor symbol. By the basis set of a configuration set $\mathscr{S}$ (in short $\mathscr{B}(\mathscr{S})$) we mean configurations in $\mathscr{S}$ with the configuration marker not being situated in the leftmost position of their right hand sides. A closure set of $\mathscr{S}$ (briefly $\mathscr{B}^*(\mathscr{S})$) consists of configurations in $\mathscr{S}$ not included in the basis set $\mathscr{B}(\mathscr{S})$. If $\mathscr{S}$ is an arbitrary set of configurations, then $\mathrm{CL}(\mathscr{S})$, the closure of $\mathscr{S}$, denotes the smallest set of configurations containing $\mathscr{S}$ with the property:

$$\text{if } A \to \alpha\,.\,B\beta \in \mathrm{CL}(\mathscr{S}), \text{ then } B \to .\,\delta \in \mathrm{CL}(\mathscr{S}) \text{ for all } B \to \delta \in P\,.$$

Let A be an alphabet symbol and $\mathscr{S}$ be a configuration set. SUCC $(\mathscr{S}, A)$ is said to be A – successor of $\mathscr{S}$ if SUCC $(\mathscr{S}, A)$ coincides with the set

$$\mathrm{CL}\left(\{B \to \alpha A \,.\, \beta \mid B \to \alpha \,.\, A\beta \in \mathscr{S}\}\right).$$

In order to accept strings from V^*, the function SUCC can be extended in a natural manner:

i. SUCC $(\mathscr{S}, \varepsilon) = \mathscr{S}$;

ii. SUCC $(\mathscr{S}, \alpha A) =$ SUCC (SUCC $(\mathscr{S}, \alpha)$, A).

The set $\mathscr{C}_G$ of all configuration sets valid for G is defined as follows:

1. the initial configuration set $\{S' \to . \perp S \perp\}$ belog to $\mathscr{C}_G$;
2. if $A \in V$, $\mathscr{S} \in \mathscr{C}_G$, then SUCC $(\mathscr{S}, A) \in \mathscr{C}_G$;
3. $\mathscr{C}_G$ contains no other configuration sets except those constructed by steps 1, 2.

Elements of the set $\mathscr{C}_G$ are said to be states of DeRemer's parsing algorithm. In conjuction with the finite state automaton defined by the set $\mathscr{C}_G$ the parsing algorithm uses a stack containing state numbers. If the parsing algorithm is in the state $\mathscr{S}$ and the input symbol is A, then it changes the state from $\mathscr{S}$ to SUCC $(\mathscr{S}, A)$ and it puts SUCC $(\mathscr{S}, A)$ on the top of the stack. If the parsing algorithm perform the reduction relative to the production $A \to \alpha$, then it pops $|\alpha|$ states from the top of the stack (now $\mathscr{S}_1$ is a new top state) and it puts SUCC $(\mathscr{S}_1, A)$ on the top position.

Let $\Pi_1 : A_{\Pi_1} \to \alpha_{\Pi_1}, \ldots, \Pi_p : A_{\Pi_p} \to \alpha_{\Pi_p}$ be labeled productions of a CF grammar G and $\#(\Pi_1), \ldots, \#(\Pi_p)$ be distinct symbols not in V. Then a reduced characteristic grammar $G' = \langle V', \Sigma', P', S' \rangle$ of G is constructed in the following way:

$$V' - \Sigma' = \{A' \mid A \in V - \Sigma\},$$

$$\Sigma' = V \cup \{\#(\Pi_1), \ldots, \#(\Pi_p)\},$$

$$P' = \{A'_{\Pi_k} \to \alpha_{\Pi_k} \# (\Pi_k) \mid A_{\Pi_k} \to \alpha_{\Pi_k} \in P\} \cup$$

$$\{A' \to \alpha B' \mid A \to \alpha B \beta, \alpha, \beta \in V^*, \; B \in V - \Sigma\}.$$

Evidently the grammar G' generates regular language and following DeRemer [1] it is exactly the language accepted by DR automaton $\mathscr{A}_G$. Hence the automaton $\mathscr{A}_G$ abstracts the essential structure of DeRemer's parsing algorithm.

A state $\mathscr{S}$ of the automaton $\mathscr{A}_G$ is called inadequate if it contains at least two configurations one of which is of the form $A_{\Pi_1} \to \alpha_{\Pi_1}$. . Thus there occurs a transition from $\mathscr{S}$ both under the symbol $\#(\Pi_1)$ of the configuration Π_1 and under successor symbols of other configurations in $\mathscr{S}$. If such a case is always solvable by means of k – look – ahead sets [1], then the grammar is SLR (k).

Example 1: Consider a grammar G_1 with productions

$$\Pi_1 : S \to \perp E \perp$$
$$\Pi_2 : E \to E + T$$
$$\Pi_3 : E \to T$$
$$\Pi_4 : T \to T \times F$$
$$\Pi_5 : T \to F$$
$$\Pi_6 : F \to a$$
$$\Pi_7 : F \to (E)$$

Grammar G_1 is SLR (1) grammar and its DR automaton is given in Fig. 1.

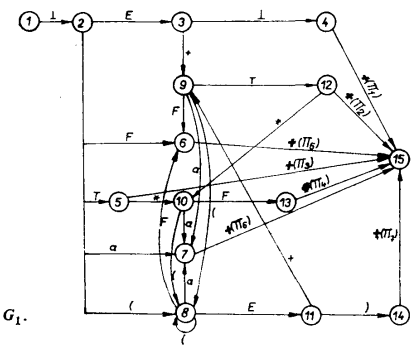

Fig. 1. DR automaton for grammar G_1.

4. ESTIMATION OF THE NUMBER OF STATES

The investigation of the number of states is greatly facilitated by the introduction of the method for partitioning of grammar rules into not necessary disjoint sets. In certain cases the upper estimation given as a consequence of the mentioned method can be improved by means of left recursive symbols.

Let $\mathcal{U}$ be the totality of configurations in the closure set of a state $\mathcal{S}$ with the same successor symbol. The set is characterized by means of corresponding set of derivation symbols $\mathcal{A}_{\mathcal{U}} = \bigcap_{\Pi \in \mathcal{U}} \mathcal{A}_{\Pi}$, where the symbol $\mathcal{A}_{\Pi}$ for some production $\Pi : A_{\Pi} \to \alpha_{\Pi}$ denotes the system of sets of infix nonterminal symbols $E = \{B_1, \ldots, B_m\}$ with the properties:

1. for $1 \leqq i \leqq m$ there exists $w_i \in V^*$ and P contains a production $C_i \to \alpha_i B_i \beta_i$, $\alpha_i, \beta_i \in V^*$ such that $\alpha_i = \delta\alpha_{i+1}$ holds for $\delta \in V^*$

a) if $\delta \neq \varepsilon$, then $S \Rightarrow_R^* w_i C_i \beta \Rightarrow_R w_i \delta C_{i+1} \gamma$ holds for $\beta, \gamma \in V^*$;

b) if $\delta = \varepsilon$, then $S \Rightarrow_R^* w_p C_p \gamma_p$ holds for $\gamma_p \in V^*$, $p = i, i+1$, $w_i = w_{i+1}$;

2. if $A \in E$, then $A_\Pi \in \mathscr{L}(A)$;

3. if for an infix nonterminal symbol B_k there exists $w_k \in V^*$ and a rule $C_k \to \alpha_k B_k \beta_k$, $\alpha_k, \beta_k \in V^*$ such that

a) at least for one $A \in E$ and B_k either 1a or 1b holds,

b) $A_\Pi \in \mathscr{L}(B_k)$

then $B_k \in E$.

The set $\mathscr{R}$ is defined to be the system of all sets $\mathscr{U}$ with the property $\mathscr{A}_\mathscr{U} \neq \emptyset$. We proceed with the formal definition.

Definition 1. A set $\mathscr{R}$ is defined as a system of all sets $\mathscr{U} \subset P$ with the following properties

i. right parts of all productions from $\mathscr{U}$ begin with the same symbol from V;

ii. $\bigcap_{\Pi \in \mathscr{U}} \mathscr{A}_\Pi \neq \emptyset$ (i.e. there exists a system of derivation sets and from each of them for every production $\Pi \in \mathscr{U}$ a word beginning with the symbol A_Π can be derived);

iii. if the right − hand side of some rule Π_1 begins with the same symbol as the right − hand side of rules from $\mathscr{U}$ and it holds $\mathscr{A}_{\Pi_1} \subset \bigcap_{\Pi \in \mathscr{U}} \mathscr{A}_\Pi$ then $\Pi_1 \in \mathscr{U}$.

Condition iii secures that $\mathscr{R}$ consists of maximal sets of grammar rules satisfying the conditions i and ii.

Not all configurations of some state $\mathscr{S}$ with the same successor symbol are contained in the closure set of $\mathscr{S}$. Such a case can occur if a successor symbol of some configuration from $\mathscr{B}(\mathscr{S})$ is left recursive. The set $\mathscr{D}$ is determined by means of a system of configuration sets with the same successor symbol simultaneously contained in the basis set $\mathscr{B}(\mathscr{S})$ and the closure set $\mathscr{B}^*(\mathscr{S})$ of some state $\mathscr{S}$.

Definition 2. Let $\mathscr{K}_A$ denote the set $\{A\alpha \mid A_\Pi \to \beta A \alpha \text{ from } P, \beta \in V^+\}$. Then

$$\mathscr{D} = \{\mathscr{U}_1^* \cap \mathscr{U}_2^* \mid \text{there exists } \mathscr{U}_1 \in \mathscr{R} \text{ and a left recursive symbol } A \in V \text{ with the property } \bigcap_{\Pi \in \mathscr{U}_1} \mathscr{A}_\Pi = \{\{A\}\} \text{ such that } \mathscr{U}_1^* = \bigcup_{\Pi \in \mathscr{U}_1} H(\alpha_\Pi, |\alpha_\Pi|) \text{ and } \mathscr{U}_2^* = \bigcup_{\beta \in \mathscr{K}_A} H(\beta, |\beta|)\} .$$

Theorem 1. If G is a reduced CF grammar and $\mathscr{P}$ is the number of states of DR automaton $\mathscr{A}_G$, then

$$\mathscr{P} \leqq 2 + \sum_{\mathscr{U} \in \mathscr{R}} p\Big(\bigcup_{\Pi \in \mathscr{U}} H(\alpha_\Pi, |\alpha_\Pi|)\Big) - p\Big(\bigcup_{\mathscr{V} \in \mathscr{D}} \mathscr{V}\Big) .$$

Proof. Consider a set $\mathscr{M}(A, \mathscr{S})$ to be the totality of all configurations in $\mathscr{S}$ with the successor symbol A. Three cases can be distinguished:

i. $\mathscr{M}(A, \mathscr{S}) \subset \mathscr{B}(\mathscr{S})$,

ii. $\mathscr{M}(A, \mathscr{S}) \subset \mathscr{B}^*(\mathscr{S})$ ($\mathscr{M}(A, \mathscr{S})$ is of the type 1),

iii. $\mathscr{M}(A, \mathscr{S}) \cap \mathscr{B}(\mathscr{S}) \neq \emptyset$, $\mathscr{M}(A, \mathscr{S}) \cap \mathscr{B}^*(\mathscr{S}) \neq \emptyset$ ($\mathscr{M}(A, \mathscr{S})$ is of the type 2).

Case i. $\mathscr{M}(A, \mathscr{S})$ can be expressed in the form

$$\mathscr{M}(A, \mathscr{S}) = \mathscr{M}^{(1)}(A, \mathscr{S}) \cup \mathscr{M}^{(2)}(A, \mathscr{S}) \cup \ldots \cup \mathscr{M}^{(m)}(A, \mathscr{S}) \quad m \geqq 1$$

where $\mathscr{M}^{(1)}(A, \mathscr{S})$ is derived from a configuration set of the type 1 (i.e. there exist a state $\mathscr{S}_1$, a symbol B_1 and a string α_1 such that $\mathscr{M}^{(1)}(A, \mathscr{S}) \subseteq \mathrm{SUCC}(\mathscr{S}_1, B_1\alpha_1)$ holds and $\mathscr{M}(B_1, \mathscr{S}_1)$ is of the type 1) and $\mathscr{M}^{(k)}(A, \mathscr{S})$ for $2 \leqq k \leqq m$ is derived from a configuration set of the type 2 (i.e. there exist a state $\mathscr{S}_k$, a symbol B_k and a string α_k such that $\mathscr{M}^{(k)}(A, \mathscr{S}) \subset \mathrm{SUCC}(\mathscr{S}_k, B_k\alpha_k)$ holds and $\mathscr{M}(B_k, \mathscr{S}_k)$ is of the type 2 while $|\alpha_j| > |\alpha_{j+1}|$ for $1 \leqq j \leqq m - 1$). It may be noticed that the set $\mathscr{M}(A, \mathscr{S})$ in the case i can be derived from a set $\mathscr{M}(A, \mathscr{S})$ either of the type 1 or of the type 2. Thus from $\mathscr{M}(A, \mathscr{S})$ in the case i can be derived just those configuration sets, which are derivable from sets having one of distinguishable types. This consideration enables us to subsume the case i under one of the following cases.

Case ii. For the number of states derivable from $\mathscr{M}(A, \mathscr{S})$ of the type 1 following statement can be obtained

$$p\Big(\bigcup_{\Pi \in \mathscr{M}(A,\mathscr{S})} \bigcup_{1 \leqq i \leqq |\alpha_\Pi|} \mathrm{SUCC}(\mathscr{S}, h(\alpha_\Pi, i))\Big) = p\Big(\bigcup_{\Pi \in \mathscr{M}(A,\mathscr{S})} H(\alpha_\Pi, |\alpha_\Pi|)\Big) .$$

Case iii. Let $\mathscr{M}(A, \mathscr{S})$ be of the type 2 and let

$$\mathscr{V}(A, \mathscr{S}) = \{\alpha \mid \alpha \in H(\alpha_\Pi, |\alpha_\Pi|),\ \Pi \in \mathscr{M}(A, \mathscr{S}) \cap \mathscr{B}^*(\mathscr{S})\} \cap$$

$$\cap \{\alpha \mid \alpha \in H(\alpha_\Pi, |\alpha_\Pi|),\ A_\Pi \to \alpha_{\Pi_1} . \alpha_{\Pi_2} \in \mathscr{M}(A, \mathscr{S}) \cap \mathscr{B}(\mathscr{S})\} .$$

Then the number of new states derivable from a set $\mathscr{M}(A, \mathscr{S}) \cap \mathscr{B}^*(\mathscr{S})$ is

$$p\Big(\bigcup_{\Pi \in \mathscr{M}(A,\mathscr{S}) \cap \mathscr{B}^*(\mathscr{S})} H(\alpha_\Pi, |\alpha_\Pi|)\Big) - p\Big(\bigcup_{\alpha \in \mathscr{V}(A,\mathscr{S})} \mathrm{SUCC}(\mathscr{S}, \alpha)\Big) .$$

Cases in which ii and iii can occur are investigated separately.

Proposition 1. Let $\mathscr{S}_1$ and $\mathscr{S}_2$ be different states and A be a left recursive symbol. Furthermore, let $\mathscr{M}(A, \mathscr{S}_1)$ be of the type 1 and $\mathscr{M}(A, \mathscr{S}_2)$ be of the type 2 while $\mathscr{M}(A, \mathscr{S}_1) = \mathscr{M}(A, \mathscr{S}_2) \cap \mathscr{B}^*(\mathscr{S}_2)$. Then the following relation holds

$$\bigcap_{\Pi \in \mathscr{M}(A,\mathscr{S}_1)} \mathscr{A}_\Pi \neq \{\{A\}\} .$$

Proof. Because $\mathscr{M}(A, \mathscr{S}_2)$ is the set of the type 2 and $\mathscr{M}(A, \mathscr{S}_1) = \mathscr{M}(A, \mathscr{S}_2) \cap$ $\cap\, \mathscr{B}^*(\mathscr{S}_2)$ there holds $\{\{A\}\} \subset \bigcap_{\Pi \in \mathscr{M}(A, \mathscr{S}_1)} \mathscr{A}_\Pi$. Moreover, $\mathscr{M}(A, \mathscr{S}_1)$ is of the type 1 and therefore there exists a set of nonterminal symbols $\{B_1, \ldots, B_m\}$ being different from $\{A\}$ with the property $\{B_1, \ldots, B_m\} \in \bigcap_{\Pi \in \mathscr{M}(A, \mathscr{S}_1)} \mathscr{A}_\Pi$. Thus we get $\{\{A\}\} \neq$ $\neq \bigcap_{\Pi \in \mathscr{M}(A, \mathscr{S}_1)} \mathscr{A}_\Pi$ by which the Proposition is proved.

Next, we establish the relation between members of $\mathscr{R}$ and sets of the type 1, namely,

Proposition 2. $\mathscr{M}(A, \mathscr{S}) \in \mathscr{R}$ if and only if the nonempty set $\mathscr{M}(A, \mathscr{S})$ is of the type 1.

Proof. Let $\mathscr{M}(A, \mathscr{S})$ be of the type 1 and $\mathscr{A}$ be a set of nonterminal successor symbols of configutations from $\mathscr{B}(\mathscr{S})$ with A as their 'left' symbol. The assertion $\mathscr{A} \in \bigcap_{\Pi \in \mathscr{M}(A, \mathscr{S})} \mathscr{A}_\Pi$ follows from the facts of the case i. Hence $\mathscr{M}(A, \mathscr{S}) \in \mathscr{R}$. Conversely the assertion $\mathscr{M}(A, \mathscr{S}) \in \mathscr{R}$ implies the existence of the set of derivation symbols $\bigcap_{\Pi \in \mathscr{M}(A, \mathscr{S})} \mathscr{A}_\Pi$ such that $\mathscr{M}(A, \mathscr{S}) \subset \mathscr{B}^*(\mathscr{S})$ and $A \in \mathscr{L}(B)$ hold for all derivation symbols B.

It obviously follows from the foregoing discussion that all states of DR automaton can be derived from configuration sets of the type 1 i.e. from elements of the set $\mathscr{R}$. By means of the case ii we get the 'rough' estimation in the form (1)

$$\sum_{\mathscr{U} \in \mathscr{R}} p\Big(\bigcup_{\Pi \in \mathscr{U}} H(\alpha_\Pi, |\alpha_\Pi|)\Big) .$$

If for a set $\mathscr{U} \in \mathscr{R}$ and a left recursive symbol A holds $\bigcap_{\Pi \in \mathscr{U}} \mathscr{A}_\Pi = \{\{A\}\}$, then following Proposition 1 $\mathscr{U}$ belongs to closure sets of those states the basis sets of which contain at least one configuration of the form $A_\Pi \to \alpha \,.\, A\beta$. Then from Proposition 1 follows that some states are already numbered in the above estimation (1). The total number of states counted more than once is

$$p(\mathscr{V}) = p(\{\alpha \mid \alpha \in H(\alpha_\Pi, |\alpha_\Pi|), \Pi \in \mathscr{U}\} \cap$$

$$\cap\, \{\alpha \mid \alpha \in H(\alpha_{\Pi_2}, |\alpha_{\Pi_2}|)\,,\; A_\Pi \to \alpha_{\Pi_1}\alpha_{\Pi_2} \in P\,,\; \alpha_{\Pi_1} \neq \varepsilon\}) .$$

Furthermore, initial and final states have to be added. Thus the proof of Theorem 1 is completed. □

The theorem is demonstrated on two examples.

Example 2. For the grammar G_1 the set $\mathscr{R}$ is

$$\mathscr{R} = \{\{\Pi_1\}, \{\Pi_2\}, \{\Pi_3, \Pi_4\}, \{\Pi_4\}, \{\Pi_5\}, \{\Pi_6\}, \{\Pi_7\}\}$$

and

$$\sum_{\mathscr{U}\in\mathscr{R}} p\Big(\bigcup_{\Pi\in\mathscr{U}} H(\alpha_\Pi, |\alpha_\Pi|)\Big) = 17 .$$

G_1 has two left recursive symbols E, T thus $p(\bigcup_{\mathscr{V}\in\mathscr{Q}} \mathscr{V}) = 2$. Therefore the estimation given by the theorem is $\mathscr{P} \leqq 17$, while the exact number of states is 15, as can be seen in Fig. 1.

Example 3. Let G_2 be the grammar

$$\Pi_1 : S \to \bot\, E\, \bot$$
$$\Pi_2 : E \to E + T$$
$$\Pi_3 : E \to T$$
$$\Pi_4 : T \to F \uparrow T$$
$$\Pi_5 : T \to F$$
$$\Pi_6 : F \to a$$
$$\Pi_7 : F \to (E)$$

In this case

$$\mathscr{R} = \{\{\Pi_1\}, \{\Pi_2\}, \{\Pi_3\}, \{\Pi_4, \Pi_5\}, \{\Pi_6\}, \{\Pi_7\}\}$$

and

$$\sum_{\mathscr{U}\in\mathscr{R}} p\Big(\bigcup_{\Pi\in\mathscr{U}} H(\alpha_\Pi, |\alpha_\Pi|)\Big) = 14 .$$

Grammar G_2 has a left recursive symbol E, therefore $p(\bigcup_{\mathscr{V}\in\mathscr{Q}} \mathscr{V}) = 1$. The estimation given by the theorem is equal to the exact number of states of DR automaton for G_2 (cf. [1, p. 455]) $\mathscr{P} = 15$.

For special subclasses of the entire class of CF grammars the estimation given by the main theorem can be expressed in a simpler form. As the first corollary of the theorem we give the result by Král and Demner [3] about LL (1) grammars which form the largest known class of grammars with the following property: the number of states can be expressed by the sum of the length of productions increased by two. A class of all grammars satisfying this property contains properly all LL (1) grammars.

Corollary 1. Let G be a reduced ε – free LL (1) grammar. Then the number of states of the corresponding DR automation is

$$\mathscr{P} = 2 + \sum_{\Pi\in P} |\alpha_\Pi| .$$

Proof. From the paper [4] follows that LL (1) grammars are LR (1) grammars without left recursive symbols with the property that the rules forming configurations in the closure set $\mathscr{B}^*(\mathscr{S})$ of some state $\mathscr{S}$ have not the same prefix. Hence $\mathscr{R}$ consists of one-element sets and for every set $\mathscr{U} \in \mathscr{R}$ there exists a rule $\Pi_1 \in P$ such that

$$p\Big(\bigcup_{\Pi\in\mathscr{U}} H(\alpha_\Pi, |\alpha_\Pi|)\Big) = |\alpha_{\Pi_1}| .$$

According to this equality we further get the result

$$\mathscr{P} = 2 + \sum_{\mathscr{U}\in\mathscr{R}} p\Big(\bigcup_{\Pi\in\mathscr{U}} H(\alpha_\Pi, |\alpha_\Pi|)\Big) = 2 + \sum_{\mathscr{U}\in\mathscr{R}} \sum_{\Pi\in\mathscr{U}} |\alpha_\Pi| = 2 + \sum_{\Pi\in P} |\alpha_\Pi| . \qquad \square$$

Corollary 2. Let G be a reduced ε – free LL (k) grammar for $k \geqq 2$. Then the upper estimation of the number of states of the corresponding DR automaton is

$$\mathscr{P} \leqq 2 + \sum_{\mathscr{U}\in\mathscr{R}} p\Big(\bigcup_{\Pi\in\mathscr{U}} H(\alpha_\Pi, |\alpha_\Pi|)\Big) .$$

Proof. The proposition of Corollary 2 is a straightforward consequence of Theorem 1. $\square$

5. QUALITY OF ESTIMATION

The estimation of the number of states given in Theorem 1 seems to be sufficient for grammars of programming languages. We try to find out properties of the estimation in the entire class of CF grammars.

The estimation of the number of states is denoted by $\mathscr{O}(G)$ while the exact number of states of DR automaton $\mathscr{A}_G$ is denoted by $\mathscr{P}(G)$. Next, deviation of the estimation, through division $\mathscr{O}/\mathscr{P}$ in the class of CF grammars, is investigated.

Property. For each nonnegative integer n there exists a CF grammar G_n such that

$$\mathscr{O}(G_n)/\mathscr{P}(G_n) > n .$$

Proof. Consider a grammar G_n given by productions

$$\begin{aligned} S &\to A_1 A_2 \ldots A_{n+1}, \\ A_1 &\to A_2 \mid a, \\ A_2 &\to A_3 \mid a a_2^{k_2}, \\ &\vdots \\ A_n &\to A_{n+1} \mid a a_n^{k_n}, \\ A_{n+1} &\to a a_{n+1}^{k_{n+1}}. \end{aligned}$$

Numbers $\mathscr{O}$, $\mathscr{P}$ for G_n are counted

$$\begin{aligned} \mathscr{O}(G_n) &= 4 + 3n + 2k_2 + 3k_3 + \ldots + nk_n + (n+1)\,k_{n+1} , \\ \mathscr{P}(G_n) &= 4 + 3n + k_2 + \ k_3 + \ldots + \ k_n + k_{n+1} . \end{aligned}$$

For division we get

$$\mathcal{O}(G_n)/\mathcal{P}(G_n) = 1 + \frac{k_2 + 2k_3 + \ldots + nk_{n+1}}{4 + 3n + k_2 + \ldots + k_{n+1}}.$$

There exists $k_{n+1} \geqslant 4 + 3n + k_2 + \ldots + k_n$ such that

$$\frac{k_2 + 2k_3 + \ldots + nk_{n+1}}{4 + 3n + k_2 + \ldots + k_{n+1}} > n - 1$$

and thus we get the result $\mathcal{O}(G_n)/\mathcal{P}(G_n) > n$. □

We give an upper estimation of the number of states of DR automaton as a function of characteristics of grammar not depending on derivation. Such characteristics can be

lg the length of the longest production of G,

p_A the number of productions from G beginning with the symbol A.

The result is summarized in the following theorem.

Theorem 2. Let G be a reduced CF grammar. Then for the number of states $\mathcal{P}$ of DR automaton $\mathcal{A}_G$ holds

$$\mathcal{P} \leqq 2 + \sum_{A \in V} \sum_{k=1}^{p_A} \binom{p_A}{k} + (lg - 1) \sum_{A \in V} \sum_{k=1}^{p_A} \binom{p_A}{k} k.$$

Proof. The set $\mathcal{R}$ from Definition 1 can be considered as the partition of the set of productions P into not necessarily disjoint elements. The assertion of the theorem follows from the fact that the maximal number of productions of all partition elements is

$$\sum_{A \in V} \sum_{k=1}^{p_A} \binom{p_A}{k} k.$$

□

6. CONCLUSIONS

Three main features of DR automaton $\mathcal{A}_G$ must be taken into account when counting states of $\mathcal{A}_G$. They are

a) partitioning of all rules into not necessarily disjoint sets of rules;

b) left recursive symbols which enable transition of initial and not initial configurations under the same symbol;

c) merging of states, containing the same set of configurations with the same derivation sets, produced by DeRemer's method from the various states.

The upper estimation of the number of states covers both a) and b) but not point c). Deviation of the estimation from the exact number of states can be arbitrarily large for some special sample grammars due to point c).The estimation is sufficient for grammars used in practice.

(Received July 16, 1973.)

REFERENCES

[1] F. L. DeRemer: Simple LR(k) grammars. Comm. ACM *14* (July 1971), 453—460.

[2] D. E. Knuth: On the Translation of Languages from Left to Right. Information and Control *8* (1965), 607—638.

[3] J. Král, J. Demner: A Note on Number of States of the DeRemer's Recognizer. Information Processing Letters *2* (1973), 22—23.

[4] D. J. Rosekrantz, R. E. Stearns: Properties of Deterministic Top-down Grammars. Information and Control *17* (1970), 226—256.

RNDr. Peter Ružička; Výskumné výpočtové stredisko Program OSN (Computing Research Centre UN D. P.), Dúbravská cesta 3, 885 31 Bratislava. Czechoslovakia.