Iterated grammars

By L. CSIRMAZ

1. Notations and definitions

1.1. Let Σ be any, finite or infinite, set. Σ^* denotes the set of finite sequences of elements of Σ including the empty sequence which is denoted by ε. Σ^+ stands for $\Sigma^*-\{\varepsilon\}$. If $\alpha\in\Sigma^*$ then $|\alpha|$ is the length of the sequence, in particular $|\varepsilon|=0$. The elements of Σ^* are called words. The mirror image of a word α is denoted by α^{-1}.

If Σ is finite we refer it as an alphabet. The subsets of Σ^* are the languages over Σ.

If Σ does not contain the comma symbol, we define Σ^s as the set of sequences of elements of Σ separated by commas. For example, if $\Sigma=\{ab, a, b\}$ then "a, b, ab" and "ab" are elements of Σ^s. Clearly $\Sigma^*\cap\Sigma^s=\Sigma\cup\{\varepsilon\}$. If $\alpha\in\Sigma^s$ then $\|\alpha\|$ denotes the length of the sequence, i.e. the number of commas in α plus one. For example $\|a, b, ab\|=3$, $\|ab\|=1$, $\|\varepsilon\|=0$ but $|a, b, ab|=6$.

1.2. A grammar or metagrammar is a 4-tuple $\mathcal{G}=\langle N, T, P, S\rangle$ where N and T are disjoint finite sets of nonterminal and terminal symbols, respectively, P is a finite set of production rules of the form $\alpha\to\beta$ where $\alpha\in N^+$, $\beta\in(N\cup T)^*$, and $S\in N$ is the starting symbol. $\mathcal{L}(\mathcal{G})$ denotes the language generated by $\mathcal{G}$.

Grammars are classified by the structure of their production rules as it can be seen in Table 1 below. The language $L\subset T^*$ is of type τ $(=0, 1, 2, 3)$ if there is a grammar of type τ generating L. The family of languages of type τ is denoted by $\chi(\tau)$.

$\mathcal{G}$ is of type	if $\alpha\to\beta\in P$ implies
0 (phrase structure)	anyway
1 (context sensivite)	S does not occur in β and either $\|\alpha\|\leqq\|\beta\|$ or $\alpha=S$ and $\beta=\varepsilon$
2 (context free)	$\|\alpha\|=1$
3 (regular)	$\|\alpha\|=1$ and either $\|\beta\|\leqq 1$ or β is of the form tn where $t\in T$ and $n\in N$

Table 1.

1.3. An iterated grammar is a 5-tuple $\mathscr{I}=\langle \Sigma, N, T, P, S\rangle$ where Σ and T are disjoint finite sets none of them containing the symbols $\Rightarrow$ and , (double arrow and comma). $N\subseteq \Sigma^+$ is the set of nonterminal symbols, T is the set of terminal symbols. P is the set of production rules of the form $\alpha\Rightarrow\beta$ where $\alpha\in N^s-\{\varepsilon\}$, $\beta\in(N\cup T)^s$ and $S\in N$ is the starting symbol. The sets N and P may be infinite. The language $\mathscr{L}(\mathscr{I})$ generated by the iterated grammar $\mathscr{I}$ is a subset of T^* the elements of which can be derived from S in the usual way using finitely many production rules only. During the derivation the commas serve as separators between the symbols but they are abandoned at the end.

Iterated grammars are classified also as Table 2 shows.

$\mathscr{I}$ is of type	if $\alpha\Rightarrow\beta\in P$ implies
0	anyway
1	S does not occur in β and either $\|\|\alpha\|\|\leqq\|\|\beta\|\|$ or $\alpha=S$ and $\beta=\varepsilon$
2	$\|\|\alpha\|\|=1$
3	$\|\|\alpha\|\|=1$ and either $\|\|\beta\|\|\leqq 1$ or β is of the form t, w where $t\in T$ and $w\in N$

Table 2.

The iterated grammar $\mathscr{I}=\langle \Sigma, N_2, T_2, P_2, S_2\rangle$ is said to be generated by the metagrammar $\mathscr{G}=\langle N_1, T_1, P_1, S_1\rangle$ if

$$\Sigma = T_1-T_2\neq\emptyset,\quad N_2=\Sigma^+,\quad T_2\subseteq T_1,\quad P_2=\mathscr{L}(\mathscr{G}).$$

An iterated grammar is of type (σ, τ) if it is of type τ and there is a metagrammar of type σ which generates it. A language L is of type (σ, τ) if there is an iterated grammar of type (σ, τ) generating L. The family of languages of type (σ, τ) is denoted by $\chi(\sigma, \tau)$.

2. The theorems

Because every finite language is regular, and $\chi(\tau)\subseteq\chi(\tau')$ if $\tau\geqq\tau'$ we have the following

PROPOSITION. If $\sigma\geqq\sigma'$ and $\tau\geqq\tau'$ then

$$\chi(\tau)\subseteq\chi(3,\tau)\subseteq\chi(\sigma,\tau)\subseteq\chi(\sigma',\tau')\subseteq\chi(0).$$

Theorem 1. $\chi(3,\tau)=\chi(\tau)$ for $\tau=0, 1, 2, 3$.

Proof. For $\tau=0$ the Proposition implies the statement. For the other cases first we need a

LEMMA. Let $L\subseteq(T\cup\{a\})^*$ be a regular language, $a\notin T$. Then there is a finite set R, a regular language $K\subseteq R^*$ and regular languages $K_b\subseteq T^*$ for every $b\in R$ such that

$$L=\{w_1 a w_2 a\dots a w_n:\ w_i\in K_{b_i}\text{ and }b_1 b_2\dots b_n\in K\}.$$

REMARK. The converse of the Lemma is evidently true, i.e. if K and the K_b's are regular languages then L is regular, too.

Proof of the lemma. It is well-known (see, e.g., [1]) that $L-\{\varepsilon\}$ can be generated by a regular grammar $\mathcal{G}=\langle N, T\cup\{a\}, P, S\rangle$ where P consists of rules of the form $A\to x$ and $A\to xB$ only ($A, B\in N$, $x\in T\cup\{a\}$). Now define $P_0, P_1, Q_0, Q_1\subseteq P$ as follows.

$$P_0 = \{\alpha\in P\colon \alpha = A\to aB \text{ for some } A, B\in N\},$$

$$P_1 = \{\alpha\in P\colon \alpha = A\to xB \text{ for some } A, B\in N,\ x\in T\},$$

$$Q_0 = \{\alpha\in P\colon \alpha = A\to a \text{ for some } A\in N\},$$

$$Q_1 = \{\alpha\in P\colon \alpha = A\to x \text{ for some } A\in N,\ x\in T\}.$$

Obviously, $P=P_0\cup P_1\cup Q_0\cup Q_1$. Let s and f be two new symbols (for start and finish) and define

$$R = \{\langle\alpha,\beta\rangle\colon \alpha,\beta\in P_0\cup Q_0\}\cup\{\langle s,\beta\rangle\colon \beta\in P_0\cup Q_0\}\cup\{\langle\alpha, f\rangle\colon \alpha\in P_0\cup Q_0\}\cup\{\langle s, f\rangle\}.$$

The languages $K_{\langle\alpha,\beta\rangle}$ for $\langle\alpha,\beta\rangle\in R$ will be the "cuts" starting after symbol a generated by the rule α and ending before the next symbol a generated by the rule β. We need two more definitions. For $A\in N$ let

$$P_0(A) = \begin{cases}\{A\to\varepsilon\} & \text{if } A\to aB\in P_0 \text{ for some } B\in N \text{ or } A\to a\in Q_0,\\ \emptyset & \text{otherwise,}\end{cases}$$

$$P_1(A) = P_1\cup\{B\to x\colon B\to xA\in P_1\}.$$

Now we are ready to define the languages $K_{\langle\alpha,\beta\rangle}$ for all $\langle\alpha,\beta\rangle\in R$. If $\alpha\in Q_0$, $\beta\in P_0\cup Q_0$ then let $K_{\langle\alpha,\beta\rangle}=\emptyset$, $K_{\langle\alpha, f\rangle}=\{\varepsilon\}$. If $\alpha=A\to aB\in P_0$ and either $\beta=C\to aD\in P_0$ or $\beta=C\to a\in Q_0$ then $K_{\langle\alpha,\beta\rangle}$ is the language generated by the grammar $\langle N, T, P_0(B)\cup P_1(C), B\rangle$, $K_{\langle s,\beta\rangle}$ is the language generated by $\langle N, T, P_0(S)\cup P_1(C), S\rangle$ and $K_{\langle\alpha, f\rangle}$ is generated by $\langle N, T, P_0(B)\cup P_1\cup Q_1, B\rangle$. Finally, $K_{\langle s, f\rangle}$ is the language generated by $\langle N, T, P_1\cup Q_1, S\rangle$ plus the empty word if it was also in L.

What remained is to define the language K. It is the one which is generated by the grammar

$$\langle P_0\cup Q_0\cup\{s, f\}, R, P^K, s\rangle$$

where

$$P^K = \{\alpha\to\langle\alpha,\beta\rangle\beta\colon \langle\alpha,\beta\rangle\in R,\ \beta\neq f \text{ and } K_{\langle\alpha,\beta\rangle}\neq\emptyset\}\cup$$

$$\cup\{\alpha\to\langle\alpha, f\rangle\colon \langle\alpha, f\rangle\in R \text{ and } K_{\langle\alpha, f\rangle}\neq\emptyset\}.$$

It is easy to check that R, K and the K_b's satisfy the requirements. □

Now we return to the proof of the Theorem 1. Let P be the regular set of production rules of the iterated grammar $\mathcal{I}=\langle\Sigma, N, T, P, S\rangle$. In this case $P\subseteq(\Sigma\cup T\cup\{\Rightarrow\}\cup\{\ ,\ \})^*$ and neither the double arrow nor the comma is an element of $\Sigma\cup T$. The double arrow must occur exactly once in every production rule, so, by the Lemma, there are regular languages P_j^L and P_j^R over $\Sigma\cup T\cup\{\ ,\ \}$ such that P is the finite union of languages

$$\{w_1\Rightarrow w_2\colon w_1\in P_j^L,\ w_2\in P_j^R\}.$$

Applying the Lemma to the languages P_j^L and P_j^R with the comma as the special terminal symbol, we get languages K_j^L and K_j^R over disjoint alphabets R_j^L and R_j^R for each j, and finitely many regular languages K_i over $\Sigma \cup T$ indexed by the elements of $I = \bigcup_j (R_j^L \cup R_j^R)$. To be more precise the K_i's are subsets of $\Sigma^+ \cup T$.

For $w_1, w_2 \in \Sigma^+$ define the relation $w_1 \equiv w_2$ as $w_1 \in K_i \leftrightarrow w_2 \in K_i$ for all $i \in I$. It is clear that this is an equivalence relation and there are finitely many equivalence classes (no more than 2^k where k is the cardinality of I). The definition of equivalence means that if $\alpha \in P$ is a production rule, $w_1 \in \Sigma^+$ is a nonterminal symbol in it and $w_1 \equiv w_2$ then putting w_2 in places of the nonterminal occurences of w_1 in α the resulting word is in P, too. Therefore every derivation can be rewritten so that it contains at most one element from each equivalence class, i.e. only finitely many different nonterminal symbols are used. It means that the languages K_i can be assumed to be finite, or, equivalently, to have one element. This element will be denoted by $\mu(i)$.

We now have finitely many regular languages K_j^L and K_j^R over the finite set I, and a function $\mu\colon I \to (N \cup T)$ such that $S \in \text{range}(\mu)$. The set of production rules was reduced to the finite union of sets

$$\{w_1 \Rightarrow w_2\colon\ w_1 \in P_j^L,\ w_2 \in P_j^R\}$$

where

$$P_j^L = \{\mu(i_1), \mu(i_2), \ldots, \mu(i_n)\colon\ i_1 i_2 \ldots i_n \in K_j^L\},$$

$$P_j^R = \{\mu(i_1), \mu(i_2), \ldots, \mu(i_n)\colon\ i_1 i_2 \ldots i_n \in K_j^R\}.$$

Our next aim is to show that the K_j^L's are finite languages. If not, there are arbitrary long elements in K_j^L, i.e. fixing some $w_2 \in P_j^R$ there is an $x \in K_j^L$ such that $|x| > \|w_2\| + 1$. Let $w_1 \in P_j^L$ be the word belonging to x. Then $\|w_1\| = |x| > \|w_2\| + 1$ which contradicts the assumption that $\mathscr{G} \in \chi(3, \tau)$ with $\tau > 0$.

If in the languages K_j^R we replace $i \in I$ by $\mu(i)$ if $\mu(i) \in T$ then the following set of production rules

$$Q = \bigcup_j \{w_1 \to w_2\colon\ w_1 \in K_j^L,\ w_2 \in K_j^R\}$$

generates the same language as P does. Moreover if all of the rules of P are of type τ, then the same is true for Q.

Now we are able to give a finite grammar which generates the same language as $\mathscr{G}$ does. It is enough to start from Q and we may assume that $T \subseteq I$ and $I - T$ is the set of nonterminal symbols of Q.

Case $\tau = 3$. The same argument as above shows that the languages K_j^R must be finite. Therefore Q is finite and obviously of type 3.

Case $\tau = 2$. Because K_j^R is a regular language it is generated by some type 3 grammar $\mathscr{G}_j = \langle N_j, I, P_j, S_j \rangle$ where N_j and I are disjoint sets, $S_j \in N_j$ and the N_j's are disjoint for different j's. The grammar Q is of type 2 so $w \in K_j^L$ implies $w \in I$. Now take the following set of rules:

$$Q_j = \{w \to S_j\colon\ w \in K_j^L\} \cup P_j.$$

Obviously, $\bigcup_j Q_j$ is finite and of type 2 and $\mathscr{L}(Q) = \mathscr{L}(\bigcup_j Q_j)$.

Case $\tau = 1$. K_j^L is finite, so we may assume that it contains only one word, w_1^j, and let $|w_1^j| = n_j$. The lengths of the words of K_j^R are at least n_j, except if w_1^j

is the starting symbol, then K_j^R may contain the empty word, too. If we fix the first n_j symbols of the right hand side of a rule then the remaining part forms a regular language, which may be empty. There are only finitely many words of length n_j, therefore we may drop them into different sets, i.e. we arrive at

$$Q = \bigcup_j \{w_1^j \to w_2^j w\colon\ w \in K_j^R\} \cup Q^*$$

where $|w_1^j| = |w_2^j|$, K_j^R is regular, and Q^* is either empty or contains the rule $S \to \varepsilon$ only. The method of Case 2 now gives immediately a finite language of type 3 generating $\mathscr{L}(Q)$, only a little care should be taken of the empty word in K_j^R. □

REMARK. A close examination of the proof shows that given some regular metagrammar $\mathscr{G}$ and an iterated grammar $\mathscr{I}$ generated by $\mathscr{G}$, there is an effective procedure which gives from $\mathscr{G}$ and $\mathscr{I}$ a grammar $\mathscr{H}$ for which $\mathscr{L}(\mathscr{I}) = \mathscr{L}(\mathscr{H})$.

Theorem 2. $\chi(2, 3) = \chi(0)$.

Proof. By the Proposition, it is enough to prove that $\chi(2, 3) \supseteq \chi(0)$. Let $\mathscr{G} = \langle N, T, P, S\rangle$ be a type 0 grammar and assume that the comma and the double arrow are not in $N \cup T$. We give the iterated grammar of type (2, 3) simply by listing its production rules, which form evidently a context free language, or, what is more, a deterministic one.

Choose a new symbol $\tilde{t}$ for each $t \in T$ and let $\tilde{T} = \{\tilde{t}\colon t \in T\}$. Change all terminals in the production rules to their counterpart, let $\tilde{P}$ be the resulting set. Let $\Sigma = N \cup \tilde{T}$ and R a new symbol not in Σ or T. The desired iterated grammar is

$$\mathscr{I} = \langle \Sigma \cup \{R\}, (\Sigma \cup \{R\})^+, T, Q, S\rangle$$

where Q consists of

$$\begin{aligned}
&[R\alpha^{-1} \Rightarrow \alpha] && \text{for each } \alpha \in \Sigma^* \\
&\gamma\alpha\delta \Rightarrow R\delta^{-1}\beta^{-1}\gamma^{-1} && \text{for each } \gamma, \delta \in \Sigma^* \quad \text{and} \quad \alpha \to \beta \in \tilde{P} \\
&[\tilde{t}\alpha \Rightarrow t, R\alpha^{-1} && \text{for each } t \in T \quad \text{and} \quad \alpha \in \Sigma^*.
\end{aligned}$$

The production rules of $\mathscr{I}$ are of type 3, the derivations of the grammar $\mathscr{G}$ are encoded in the nonterminals of $\mathscr{I}$ in a straighforward way. □

Abstract

The definition of the programming language Algol 68 [2] raised the following problem: If a grammar is not given by some finite description but itself is a language generated by some metagrammar, what strength may the iterated grammar have? We show that a regular metagrammar does not increase the strength of the iterated grammar, but a context free metagrammar (even a deterministic one) with a regular iterated grammar has the strength of the phrase structure grammars.

MATHEMATICAL INSTITUTE OF THE
HUNGARIAN ACADEMY OF SCIENCES
REALTANODA U. 13—15.
BUDAPEST, HUNGARY
H—1053

References

[1] HOPCROFT and ULLMAN, *Formal languages and their relation to automata*, Addison-Wesley, 1967.
[2] VAN WIJNGAARDEN A, et al., Revised report on the algorithmic language Algol 68, Springer, 1976.

(Received May 31, 1979)