FINITE INDEX IN LANGUAGE THEORY

Ing. Stanislav ELBL, Doctoral Degree Programme (3)
Dept. of Information Systems, FIT, BUT
E-mail: elbl@fit.vutbr.cz

Supervised by: Dr. Alexander Meduna

ABSTRACT

Finite index restriction was already studied for variety of formal models and many results was published. This contribution discusses some new results and improves some known results. Contribution deals especially with grammar systems.

1 PRELIMINARIES

For any alphabet N, any sentence form α, $\#N(\alpha)$ denotes the number of the occurrences of symbols from N in α. For every rule p, $lhs(p)$ is the left hand side of p, $rhs(p)$ is the right hand side of p, and for any set of rules P, $LHS(P) = \{lhs(p) \mid p \in P\}$ is the set of all left hand sides of the rules in P. For any sentence form α, $subs(\alpha)$ is a set of all substrings of α.

2 GRAMMARS

A grammar is a quadruple $G = (N, \Sigma, P, S)$, where N is finite set of nonterminals, Σ is finite terminal alphabet, $S \in N$ is starting nonterminal and $P \subseteq (N \cup \Sigma)^* N (N \cup \Sigma)^* \times (N \cup \Sigma)^*$ is finite set of rules, which are usually written in the form $\alpha \to \beta$ where $\alpha \in (N \cup \Sigma)^* N (N \cup \Sigma)^*$ and $\beta \in (N \cup \Sigma)^*$. For any $x, y \in (N \cup \Sigma)^*$ we say that x directly derives y in G formally $x \Rightarrow_G y$, if there exists u, v, α, β, such that $x = u\alpha v$, $y = u\beta v$ and $\alpha \to \beta \in P$. Denote transitive and reflexive closure of $\Rightarrow_G$ as $\Rightarrow_G^*$. A language defined by G is $L(G) = \{w \mid w \in \Sigma^*, S \Rightarrow_G^* w\}$.

According to form of the rules there are 4 main families of formal languages – regular, context-free, context and recursively enumerable languages, respectively denoted L_3, L_2, L_1 and L_0. This classes form hierarchy: $L_3 \subset L_2 \subset L_1 \subset L_0$. For details see [3].

3 FINITE INDEX

Let $G = (N, \Sigma, P, S)$ be a grammar. Define an index of sentence form $\alpha \in (N \cup \Sigma)^*$ as $ind(\alpha) = \#N(\alpha)$. Informally, an index of sentence form is a number of occurrences of nonterminals in this sentence form. An index of any derivation d: $\alpha_0 \Rightarrow_G \alpha_1 \Rightarrow_G \alpha_2 \Rightarrow_G \ldots$

$\Rightarrow_G$ α_m is the maximum of indexes of all sentence forms in this derivation, $ind(d) = max(ind(w_i))$, where $0 \le i \le m$. For any word $x \in L(G)$, $D(x)$ is a set of all derivations of x in G and $ind(x) = min(ind(D(x)))$. Finally define an index of G as $ind(G) = max(ind(x))$, where $x \in L(G)$. G is of finite index k if every $x \in L(G)$ is at most of index k. For details see [5] or [6].

It is interesting to study a hierarchy of finite index languages. Denote L_{0FI}, L_{1FI}, L_{2FI}, L_{3FI} a families of recursively enumerable languages of finite index, context sensitive languages of finite index, context free languages of finite index and regular languages of finite index, respectively.

It is easy to show, that every regular language is of finite index – there is only one nonterminal in each sentence form derived from starting nonterminal and hence regular languages are of index 1. In [6] it is showed that every context sensitive language of finite index is also context free language of finite index, formally $L_{1FI} = L_{2FI}$; the generative power of context sensitive grammars is reduced in respect of all context sensitive grammars. The result concerning all recursively enumerable languages is in contrast with the previous one. In [6] it is proved that $L_{0FI} = L_0$, however the proof doesn't deal with proper terminals; the proof always establishes new improper terminals. Proper terminals of any grammar $G = (N, \Sigma, P, S)$ are all terminals which occurs in any sentence $x \in L(G)$. A modified proof, which doesn't establish new improper terminals follows.

Theorem 1

For any alphabet Σ, $|\Sigma|>1$, any grammar $G = (N, \Sigma, P, S)$, there exists grammar $G_{FI} = (N_{FI}, \Sigma, P_{FI}, S)$ of finite index such that $L(G_{FI}) = L(G)$.

Proof: Let Σ be an alphabet, $|\Sigma|>1$, and $G = (N, \Sigma, P, S)$ be a grammar. For some $n \ge 2$ define an injection, h, from $N \cup \Sigma$ to Σ^n, so that h is injective homomorphism when its domain is extended to $(N \cup \Sigma)^*$; h represents a coding of symbols by n-tuples of terminals.

Without any loss of generality, assume that $\{B, H, E, F\} \cap \Sigma = \varnothing$. Construct a new grammar, $G_{FI} = (N_{FI}, \Sigma, P_{FI}, S)$, where $N_{FI} = \{S, B, H, E, F\}$ and P_{FI} is constructed as follows:

$P_1 = \{ S \to B\ H\ h(S)\ E \}$
$P_2 = \{ H\ h(\alpha) \to H\ h(\beta) \mid \alpha \to \beta \in P \}$
$P_3 = \{ H\ h(a) \to h(a)\ H \mid a \in N \cup \Sigma \} \cup \{ h(a)\ H \to H\ h(a) \mid a \in N \cup \Sigma \}$
$P_4 = \{ B\ H \to F \}$
$P_5 = \{ F\ h(a) \to a\ F \mid a \in \Sigma \}$
$P_6 = \{ F\ E \to \varepsilon \}$
$P_{FI} = P_1 \cup P_2 \cup P_3 \cup P_4 \cup P_5 \cup P_6$

The derivation is started by production from P_1. Nonterminal H represents a "reading head". The productions from P_2 simulate a derivation step in G, since they are similar to productions of G – they work with coded symbols. The productions from P_3 provide that H can move from left to right by one coded symbol and conversely. When the simulated derivation in G is finished, current sentence form is $B\ h(\alpha)\ H\ h(\beta)\ E$, where $\alpha\beta \in \Sigma^*$ is sentence generated in G. The derivation in G_{FI} continues $B\ h(\alpha)\ H\ h(\beta)\ E \Rightarrow^* B\ H\ h(\alpha)\ h(\beta)\ E$ and then, by productions from P_4, P_5 and finally from P_6: $B\ H\ h(\alpha)\ h(\beta)\ E \Rightarrow F\ h(\alpha)\ h(\beta)\ E \Rightarrow^* \alpha\beta\ F\ E \Rightarrow \alpha\beta$. Formal proof of equivalence of G and G_{FI} is left to the reader.

Theorem 1 doesn't determine a generative power of finite index type-0 grammars

completely. The unsolved problem here is a case of type-0 grammars over one letter alphabet. It is easy to find a type-0 grammar of finite index over one letter alphabet, which doesn't contain improper terminals and generates non-context-free language; see Theorem 2. However generally this question stays unanswered in this contribution.

Theorem 2

There exists a grammar of type 0 over one-letter alphabet, which is of finite index and generates a non-context-free language.

Proof: Consider a grammar $G = (N, \Sigma, P, S)$, where $N = \{S, B, E, R, L\}$, $\Sigma = \{a\}$ and $P = \{S \to BRaE, Ra \to aaR, RE \to LE, aL \to Laa, BL \to BR, RE \to \varepsilon, BL \to \varepsilon, B \to \varepsilon, E \to \varepsilon\}$. This grammar generates a language $\{(a^2)^n \mid n \geq 0\}$, which is not context free. Formal proof of this fact is left to the reader.

4 GRAMMAR SYSTEMS

A grammar system is a structure $GS = (N, \Sigma, S, P_1, \ldots, P_n)$, where N is finite set of nonterminal symbols, Σ is finite terminal alphabet, $S \in N$ is starting nonterminal and $P_i \subseteq N \times (N \cup \Sigma)^*$, where $1 \leq i \leq n$, are finite set of rules called components. $G_i = (N, \Sigma, P_i, S)$ is called i-th grammar of GS and n is the degree of GS.

Let $u, v, w \in (N \cup \Sigma)^*$, $A \in N$ and $r = A \to u$ is a rule from P_i for any $1 \leq i \leq n$. Then we write $vAw \Rightarrow_i vuw$. Reflexive and transitive closure of $\Rightarrow_i$ is denoted $\Rightarrow_i^*$.

There are several derivation modes used in grammar systems:

Terminating derivation in i-th component, denoted $\Rightarrow_i^t$

Let $x, y \in (N \cup \Sigma)^*$. We write $x \Rightarrow_i^t y$ if and only if $x \Rightarrow_i^* y$ and there is no $z \in (N \cup \Sigma)^*$, such that $y \Rightarrow_i z$.

K-step derivation in i-th component, denoted $\Rightarrow_i^{=k}$

Let $x, y \in (N \cup \Sigma)^*$. We write $x \Rightarrow_i^{=k} y$ if and only if $x \Rightarrow_i^k y$.

At most k-step derivation in i-th component, denoted $\Rightarrow_i^{\leq k}$

Let $x, y \in (N \cup \Sigma)^*$. We write $x \Rightarrow_i^{\leq k} y$ if and only if $x \Rightarrow_i^j y$ for some $j \leq k$.

At least k-step derivation in i-th component, denoted $\Rightarrow_i^{\geq k}$

Let $x, y \in (N \cup \Sigma)^*$. We write $x \Rightarrow_i^{\geq k} y$ if and only if $x \Rightarrow_i^j y$ for some $j \geq k$.

Furthermore define the derivation in GS working in mode m for some $m \in \{=k, \leq k, \geq k, t\}$ as $x \Rightarrow_{GS}^m y$ if and only if $x \Rightarrow_i^m y$ for some $i \leq n$. In common way denote the transitive and reflexive closure of $\Rightarrow_{GS}^m$ as $\Rightarrow_{GS}^{m^*}$. A language defined by GS working in mode m, $L_m(GS) = \{x \mid x \in \Sigma^*, S \Rightarrow_{GS}^{m^*} x\}$.

In homogenous grammar system all components work in the same mode as defined above. Generally each component of system can work in different mode. Such systems are called heterogeneous grammar systems. It is clear, that every homogenous grammar system is a special case of heterogeneous system. For details about grammar systems see [4]. A grammar system of finite index is defined similarly to a grammar of finite index.

Theorem 3

Every grammar system of finite index over one letter alphabet generates regular language.

Proof: Let $G_S = (N_S, \{a\}, <S>, P_{S1}, \ldots P_{Sn})$ be a grammar system of finite index f with n components. Denote $\Sigma = \{a\}$. Define a homomorphism μ: $(N_S \cup \Sigma) \to (N_S \cup \{\varepsilon\})$, such that $\mu(A) = A$ for $A \in N_S$, $\mu(a) = \varepsilon$. Further define a homomorphism σ: $(N_S \cup \Sigma) \to (\Sigma \cup \{\varepsilon\})$, such that $\sigma(A) = \varepsilon$ for $A \in N_S$, $\sigma(a) = a$. Extend the domains of μ and σ to $(N_S \cup \Sigma)^*$. Informally the homomorphisms σ and μ compute a string without nonterminals and terminals, respectively.

Construct a context free grammar $G = (N, \{a\}, <S>, P)$, where:

$N = N_0 \cup N_1 \cup \ldots \cup N_n$
$N_0 = \{<A_1 \ldots A_m> \mid m \le f, A_i \in N_S, 0 \le i \le m\}$
$P = \{<> \to \varepsilon\} \cup P_1 \cup P_2 \cup \ldots \cup P_n$

Construction of N_i and P_i, $1 \le i \le n$, depends on the mode of *i-th* component and is described below. Denote $W = \{w \mid w \in (N_S \cup \Sigma)^*, \#N_S(w) \le f\}$.

Terminating derivation (t-mode)

$N_i = \{<x>_i \mid <x> \in N_0\}$
$P_i = \{<x> \to <x>_i \mid <x> \in N_0\}$
$\cup \{<vAw>_i \to \sigma(u)<v\mu(u)w>_i \mid A \to u \in P_{Si}, <vAw>_i, <v\mu(u)w>_i \in N_i\}$
$\cup \{<x>_i \to <x> \mid <x> \in N_0, subs(x) \cap LHS(P_{Si}) = \varnothing\}$

K-step derivation (=k)

$N_i = \{<x>_{i,l} \mid <x> \in N_0, 0 \le l \le k\}$
$P_i = \{<x> \to <x>_{i,0} \mid <x> \in N_0\}$
$\cup \{<vAw>_{i,l} \to \sigma(u)<v\mu(u)w>_{i,l+1} \mid$
$\mid A \to u \in P_{Si}, <vAw>_{i,l}, <v\mu(u)w>_{i,l+1} \in N_i, l < k\}$
$\cup \{<x>_{i,k} \to <x> \mid <x> \in N_0\}$

At most k-step derivation (≤k)

$N_i = \{<x>_{i,l} \mid <x> \in N_0, 0 \le l \le k\}$
$P_i = \{<x> \to <x>_{i,0} \mid <x> \in N_0\}$
$\cup \{<vAw>_{i,l} \to \sigma(u)<v\mu(u)w>_{i,l+1} \mid$
$\mid A \to u \in P_{Si}, <vAw>_{i,l}, <v\mu(u)w>_{i,l+1} \in N_i, l < k\}$
$\cup \{<x>_{i,l} \to <x> \mid <x> \in N_0 , l \le k\}$

At least k-step derivation (≥k)

$N_i = \{<x>_{i,l} \mid <x> \in N_0, 0 \le l \le k\}$
$P_i = \{<x> \to <x>_{i,0} \mid <x> \in N_0\}$
$\cup \{<vAw>_{i,l} \to \sigma(u)<v\mu(u)w>_{i,l+1} \mid$
$\mid A \to u \in P_{Si}, <vAw>_{i,l}, <v\mu(u)w>_{i,l+1} \in N_i, l < k\}$
$\cup \{<vAw>_{i,k} \to \sigma(u)<v\mu(u)w>_{i,k} \mid A \to u \in P_{Si}, <vAw>_{i,k}, <v\mu(u)w>_{i,k} \in N_i\}$
$\cup \{<x>_{i,k} \to <x> \mid <x> \in N_0\}$

Constructed regular grammar generates identical language as G_S does. All nonterminals in every sentence form are joined together in one nonterminal enclosed in brackets. Since G_S is of finite index, the number of new nonterminals is finite. In addition the nonterminals are

indexed by a number of component. Construction of set of rules assures that the derivation is simulated properly. Rigorous proof is left to the reader.

However theorem 3 deals with grammar systems over one letter alphabet only. In my future work I want investigate more general result. I suspect that the generative power of grammar systems of finite index is equal to generative power of matrix grammars of finite index (in fact I am sure by this fact and I am working on the proof).

Clearly there exist languages generated by grammar systems of finite index, which are not context-free. An example of such a system follows (without rigorous proof again):

$G_S = (\{S, A, B, X, Y\}, \{a\}, S, P_1, P_2, P_3)$, where

$P_1 = \{S \to AB, X \to A, Y \to B\}$

$P_2 = \{A \to aX, B \to bYc\}$

$P_3 = \{X \to \varepsilon, Y \to \varepsilon\}$

This grammar system working in terminating mode generates language $a^n b^n c^n$, where $n \geq 1$, which is well-known context sensitive language. It is easy to find a matrix grammar of finite index generating this language.

As suggested by example above, the generative power of grammar systems of finite index is greater then the generative power of context free grammars of finite index. More accurate relationship between these classes of languages should be a subject of research too. I suspect (in this case it is only conjecture) that there is no context-free language which is not of finite index and which can be generated by grammar system of finite index.

REFERENCES

[1] Elbl, S.: Grammar Systems of Finite Index, prepared to publication

[2] Ginsburg, S., Spanier, E.: Derivation-bounded Languages, J. Comput. System Sci. 2, page 228-250, 1968

[3] Meduna, A.: Automata and Languages: Theory and Applications, Springer, London, 2000 , ISBN 1-85233-074-0

[4] Rozenberg, G., Salomaa, A.: Handbook of Formal Languages, Volume 2, Springer, New York, 1997, ISBN 3-540-60648-3

[5] Salomaa, A.: Formal Languages, Academic Press Professional, San Diego, 1987, ISBN 0126157502

[6] Vermeir, D.: Over Strukturele Restrikties Op ETOL Systemen, Wilrijk, 1978