K-LIMITED ERASING PERFORMED BY REGULAR-CONTROLLED CONTEXT-FREE GRAMMARS

Petr Zemek
Master Degree Programme (2), FIT BUT
E-mail: xzemek02@stud.fit.vutbr.cz

Supervised by: Alexander Meduna
E-mail: meduna@fit.vutbr.cz

ABSTRACT

A regular-controlled context-free grammar erases its nonterminals in a k-limited way, where $k \geq 0$, if in every sentential form x of any successful derivation x contains at most $k|x|/(k+1)$ nonterminals from which it does derive the empty string, where $|x|$ is the length of x. This paper demonstrates that any regular-controlled context-free grammar that erases its nonterminals in this way can be converted to an equivalent regular-controlled context-free grammar without any erasing rules, while it is not known whether this is possible in general.

1 INTRODUCTION

It is a very well known fact that we can convert any context-free grammar with erasing rules to an equivalent context-free grammar without erasing rules [1]. However, whether erasing rules can be eliminated from regulated grammars in general, is an open problem [2]. This paper studies this problem in terms of regular-controlled context-free grammars and presents a condition, called k-limited erasing. This condition guarantees that if a regular-controlled context-free grammar satisfies it, we can convert this grammar to an equivalent regular-controlled context-free grammar without erasing rules.

2 PRELIMINARIES AND DEFINITIONS

This paper assumes that the reader is familiar with the formal language theory (see [1]). For a set, Q, $card(Q)$ denotes the cardinality of Q. For an alphabet, V, V^* represents the free monoid generated by V under the operation of concatenation. Let ε be the unit of V^* and $V^+ = V^* - \{\varepsilon\}$. Given a word, $w \in V^*$, $|w|$ denotes the length of w. Symbols $\langle$, $\rangle$, $\lfloor$, $\rfloor$, $\lceil$, and $\rceil$ are used to clearly unite more symbols into a single compound symbol.

A *context-free grammar* (see [1]) is a quadruple, $G = (V, T, P, S)$, where V is the *total alphabet*, $T \subset V$ is the alphabet of *terminal symbols*, $N = V - T$ is the alphabet of *nonterminal symbols*, $S \in N$ is the *start symbol*, and $P \subseteq N \times V^*$ is a finite relation, called the set of *rules*. Each rule $(A, y) \in P$ is written as $A \to y$ throughout this paper. If $u, v \in V^*$ and $A \to y \in P$, then $uAv \Rightarrow uyv$ in G according to $A \to y$. Let $\Rightarrow^*$ denote the reflexive-transitive closure of $\Rightarrow$. The *language of* G is denoted as $L(G)$ and defined as $L(G) = \{w \,|\, w \in T^*, S \Rightarrow^* w\}$. G is said to be ε-*free*

if every rule $A \to y \in P$ satisfies $y \in V^+$. Rules of the form $A \to \varepsilon$ are called *erasing rules* or, more briefly, ε-*rules*. If every $A \to y \in P$ implies $y \in T(N \cup \{\varepsilon\})$, then G is a *regular grammar*. For any $A \Rightarrow^* x$, where $A \in V$, $x \in V^*$, $\Delta(A \Rightarrow^* x)$ denotes its corresponding derivation tree (regarding derivation trees and related notions, we use the terminology of [1]). A derivation subtree whose frontier is ε is called an ε-*subtree*.

Let $G = (V,T,P,S)$ be a context-free grammar. Let Ψ be a set of symbols called *rule labels* such that $card(\Psi) = card(P)$, and ψ be a bijection from P to Ψ. For simplicity and brevity, to express that ψ maps a rule $A \to x \in P$ to r, where $r \in \Psi$, we write $r\colon A \to x \in P$; in other words, $r\colon A \to x$ means $\psi(A \to x) = r$. The symbols A and x represent the *left-hand side* of r, denoted by $lhs(r)$, and the *right-hand side* of r, denoted by $rhs(r)$, respectively. Let P^* denote the set of all sequences of rules from P. By analogy with strings from V^*, we omit all separating commas in these sequences, so we write $r_1 r_2 \dots r_n$ instead of $r_1, r_2, \dots, r_n$, where $r_i \in P$, for all $1 \leq i \leq n$, for some $n > 0$ ($n = 0$ means $r_1 r_2 \dots r_n = \varepsilon$). In the standard way, we extend ψ from P^* to Ψ^*—that is, $\psi(\varepsilon) = \varepsilon$, and $\psi(r_1 r_2 \dots r_n) = \psi(r_1)\psi(r_2)\dots\psi(r_n)$, where $n \geq 1$. Let $w_0, w_1, \dots, w_n$ be a sequence, where $w_i \in V^*$, for all $0 \leq i \leq n$, for some $n \geq 0$. If $w_{j-1} \Rightarrow w_j$ in G according to a rule $r_j \in P$, for $1 \leq j \leq n$, then we write $w_0 \Rightarrow^* w_n\ [\psi(r_1 r_2 \dots r_n)]$.

For any context-free grammar G, we automatically assume that V, N, T, S, P, and Ψ (with possible subscript G) denote its total alphabet, the alphabet of nonterminal symbols, the alphabet of terminal symbols, the start symbol, the set of rules, and the set of rule labels, respectively.

A *regular-controlled context-free grammar* (see [2]) is a pair, $R = (G, \Xi)$, where $G = (V,T,P,S)$ is a context-free grammar and $\Xi \subseteq \Psi^*$ is a regular language. The *language generated by G with control language Ξ* is denoted by $L(G,\Xi)$ and defined as $L(G,\Xi) = \{w \mid w \in T^*, S \Rightarrow^* w\ [\alpha] \text{ with } \alpha \in \Xi\}$.

Let k be a non-negative integer. Grammar G with control language Ξ *erases its nonterminals in a k-limited way* provided that it satisfies this implication: if $S \Rightarrow^* y$ in G is a derivation of the form $S \Rightarrow^* x \Rightarrow^* y$, where $x \in V^+$ and $y \in L(G,\Xi) - \{\varepsilon\}$, then in $\Delta(S \Rightarrow^* y)$, there are at most $k|x|/(k+1)$ ε-subtrees rooted at the symbols of x.

3 MAIN RESULT

Algorithm 1. *Elimination of ε-rules from any regular-controlled context-free grammar that erases its nonterminals in a k-limited way.*

Input: A context-free grammar, $G = (V_G, T_G, P_G, S_G)$, and a regular grammar, $H = (V_H, T_H, P_H, S_H)$, such that G with control language $L(H)$ erases its nonterminals in a k-limited way.

Output: An ε-free context-free grammar, $O = (V_O, T_O, P_O, S_O)$, and a regular grammar, $Q = (V_Q, T_Q, P_Q, S_Q)$, such that $L(O, L(Q)) = L(G, L(H)) - \{\varepsilon\}$.

Method: Without any loss of generality, assume that $Z \notin (V_H \cup \Psi_O)$. Initially, set $k' = k + max(\{|rhs(r)| \mid r \in \Psi_G\})$, $T_O = T_G$, $V_O = T_O \cup \{\langle X, y\rangle \mid X \in V_G, y \in N_G^*, 0 \leq |y| \leq k'\}$, $S_O = \langle S_G, \varepsilon\rangle$, $\Psi_O = \{\lfloor \langle a, \varepsilon\rangle \to a \rfloor \mid a \in T_G\}$, $P_O = \{\lfloor \langle a, \varepsilon\rangle \to a \rfloor\colon \langle a, \varepsilon\rangle \to a \mid a \in T_G\}$, $T_Q = \Psi_O$, $V_Q = T_Q \cup N_H \cup \{Z\}$, $S_Q = S_H$, and $P_Q = \{Z \to \lfloor \langle a, \varepsilon\rangle \to a \rfloor B \mid B \in \{Z, \varepsilon\}, a \in T_G\}$.

Now, repeat (1) through (3), given next, until none of the sets $\Psi_O, P_O, T_Q, N_Q, P_Q$ can be extended in this way.

(1) **If** $r\colon A \to x_0X_1x_1X_2x_2\dots X_nx_n \in P_G$ and $\langle A,w\rangle, \langle X_1, wx_0x_1\dots x_n\rangle \in N_O$, where $X_i \in V_G$, for all $1 \leq i \leq n$, $x_j \in N_G^*$, for all $0 \leq j \leq n$, $w \in N_G^*$, for some $n \geq 1$
then add $s = \lfloor r, x_0, X_1x_1, X_2x_2, \dots, X_nx_n \rfloor$ to Ψ_O and to T_Q; add $s\colon \langle A,w\rangle \to \langle X_1, wx_0 x_1\dots x_n\rangle\langle X_2,\varepsilon\rangle\dots\langle X_n,\varepsilon\rangle$ to P_O; for each $B \to r \in P_H$, add $B \to sZ$ to P_Q; for each $B \to rC \in P_H$, $C \in N_H$, add $B \to sC$ to P_Q.

(2) **If** $r\colon A \to w \in P_G$ and $\langle X, uAv\rangle, \langle X, uwv\rangle \in N_O$, where $X \in V_G$, $u,v,w \in N_G^*$
then add $s = \lfloor \langle X, uAv\rangle, r \rfloor$ to Ψ_O and to T_Q; add $s\colon \langle X,uAv\rangle \to \langle X,uwv\rangle$ to P_O; for each $B \to r \in P_H$, add $B \to sZ$ to P_Q; for each $B \to rC \in P_H$, add $B \to sC$ to P_Q.

(3) **If** $\langle X,uAv\rangle$, $\langle Y,w\rangle, \langle Y,wA\rangle \in N_O$, where $X,Y \in V_G$, $A \in N_G$, $u,v,w \in N_G^*$
then add $r = \lfloor \langle X,uAv\rangle, \langle Y,w\rangle \rfloor$ and $s = \lfloor \langle X,uv\rangle, \langle Y,wA\rangle \rfloor$ to Ψ_O and to T_Q; add $r\colon \langle X,uAv\rangle \to \langle X,uv\rangle$ and $s\colon \langle Y,w\rangle \to \langle Y,wA\rangle$ to P_O; for each $B \in N_H$, add $C = \lceil B, \langle X,uAv\rangle, \langle Y,w\rangle \rceil$ to N_Q and add $B \to rC$ and $C \to sB$ to P_Q.

Basic Idea. The resulting grammar, O, uses compound nonterminals of the form $\langle X,y\rangle$, where X is a symbol that is not erased during the derivation and y is a string of nonterminals that are erased during the derivation. The length of y is limited to $k' = k + p$, where p is the length of the longest right-hand side of a rule from P_G.

Rules introduced in (1) are used to simulate a derivation step in G in which one nonterminal is rewritten to a string of symbols, where at least one of them is not erased during the derivation. On the other hand, rules introduced in (2) are used to simulate a derivation step in G in which some to-be-erased nonterminal is rewritten to a string of to-be-erased nonterminals or ε (this rewrite is done in the second component). Since there might not be enough space to do such rewrite, rules introduced in (3) are used to move nonterminals between the second components. Because we do not need to keep any context information, it does not matter where in a sentential form they occur. In addition, as G erases its nonterminals in a k-limited way, there is always enough space to accommodate all these to-be-erased nonterminals. At the very end of any successful derivation, rules of the form $\langle a,\varepsilon\rangle \to a$, for all $a \in T_G$, are used to obtain terminal symbols from compound nonterminals.

4 CONCLUSION

Algorithm 1 represents a partial solution to the problem concerning the effect of erasing rules to the generative power of regular-regulated context-free grammars. Indeed, if these grammars erase their nonterminals in a k-limited way, they are equally powerful with or without erasing rules. Consequently, to solve this problem completely, the formal language theory can restrict its attention only to grammars that do not erase their nonterminals in this way because if they do, the present paper has answered the problem. Due to the requirements imposed on the length of this paper, the proof that Algorithm 1 is correct is omitted.

Acknowledgement: This work was partially supported by the BUT FIT grant FIT-S-10-2 and the research plan MSM0021630528.

REFERENCES

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

[2] C. Martín-Vide and V. Mitrana and G. Păun, editors. Formal Languages and Applications. Springer, 2004. ISBN 3-540-20907-7.