Abstract

The descriptional complexity of semi-conditional grammars is studied. It is proved that every recursively enumerable language is generated by a semi-conditional grammar of degree \((2, 1)\) with no more than seven conditional productions and eight nonterminals.

Key words: formal languages, semi-conditional grammars, descriptional complexity
1991 MSC: 68Q19

1 Introduction

Semi-conditional grammars (see [3,7,8]) are context-free grammars, in which two strings, called a permitting and a forbidding context, are attached to each production. Such a production is applicable if its permitting context occurs in the current sentential form while its forbidding context does not. Simple semi-conditional grammars represents a straightforward simplification of semi-conditional grammars, in which each production has just one attached string—either a permitting or a forbidding context.

Email addresses: masopust@fit.vutbr.cz (Tomáš Masopust), meduna@fit.vutbr.cz (Alexander Meduna).

Preprint submitted to Elsevier 10 April 2007
The formal language theory has discussed the descriptional complexity of simple semi-conditional grammars in detail (see [4,6,7,9]). In [7], it is proved that every recursively enumerable language is generated by a (simple) semi-conditional grammar of degree (2,1) with no more than twelve conditional productions and thirteen nonterminals. Later, in [9], this result was improved by demonstrating that every recursively enumerable language is generated by a (simple) semi-conditional grammar of degree (2,1) with no more than ten conditional productions and twelve nonterminals. Finally, this result was improved in [4] by demonstrating that every recursively enumerable language is generated by a (simple) semi-conditional grammar of degree (2,1) with no more than nine conditional productions and ten nonterminals.

This paper discusses the descriptional complexity of semi-conditional grammars because this topic has not been studied at all so far. It demonstrates stronger results about this complexity for them than the above results for simple semi-conditional grammars. Specifically, it proves that every recursively enumerable language is generated by a semi-conditional grammar of degree (2,1) with no more than seven conditional productions and eight nonterminals.

2 Preliminaries and Definitions

This paper assumes that the reader is familiar with the theory of formal languages (see [1,5]). For an alphabet \( V \), \( V^* \) represents the free monoid generated by \( V \). The unit of \( V^* \) is denoted by \( \varepsilon \). Set \( V^+ = V^* - \{ \varepsilon \} \). Set \( \text{sub}(w) = \{ u : u \text{ is a substring of } w \} \).

In [2], it was shown that every recursively enumerable language is generated by a grammar

\[
G = (\{ S, A, B, C \}, T, P \cup \{ ABC \rightarrow \varepsilon \}, S)
\]

in the Geffert normal form, where \( P \) contains context-free productions of the form

\[
\begin{align*}
S & \rightarrow uSa, \quad \text{where } u \in \{ A, AB \}^*, a \in T, \\
S & \rightarrow uSv, \quad \text{where } u \in \{ A, AB \}^*, v \in \{ BC, C \}^*, \\
S & \rightarrow uv, \quad \text{where } u \in \{ A, AB \}^*, v \in \{ BC, C \}^*.
\end{align*}
\]

In addition, any terminal derivation is of the form

\[
S \Rightarrow^* w_1w_2w
\]
by productions from $P$, where $w_1 \in \{A, B\}^*$, $w_2 \in \{B, C\}^*$, $w \in T^*$, and

$$w_1w_2w \Rightarrow^* w$$

by $ABC \to \varepsilon$.

A semi-conditional grammar, $G$, is a quadruple

$$G = (N, T, P, S),$$

where

- $N$ is a nonterminal alphabet;
- $T$ is a terminal alphabet such that $N \cap T = \emptyset$;
- $S \in N$ is the start symbol; and
- $P$ is a finite set of productions of the form

$$(X \to \alpha, u, v)$$

with $X \in N$, $\alpha \in (N \cup T)^*$, and $u, v \in (N \cup T)^+ \cup \{0\}$, where $0 \not\in N \cup T$ is a special symbol.

If $u \neq 0$ or $v \neq 0$, then the production $(X \to \alpha, u, v) \in P$ is said to be conditional. $G$ has degree $(i, j)$ if for all productions $(X \to \alpha, u, v) \in P$, $u \neq 0$ implies $|u| \leq i$ and $v \neq 0$ implies $|v| \leq j$. For $x \in (N \cup T)^+$ and $y \in (N \cup T)^*$, $x$ directly derives $y$ according to the production $(X \to \alpha, u, v) \in P$, denoted by

$$x \Rightarrow y$$

if $x = x_1Xx_2$, $y = x_1\alpha x_2$, for some $x_1, x_2 \in (N \cup T)^*$, and $u \neq 0$ implies that $u \in \text{sub}(x)$ and $v \neq 0$ implies that $v \not\in \text{sub}(x)$. As usual, $\Rightarrow$ is extended to $\Rightarrow^i$, for $i \geq 0$, $\Rightarrow^+$, and $\Rightarrow^*$. The language generated by a semi-conditional grammar, $G$, is defined as

$$\mathcal{L}(G) = \{w \in T^* : S \Rightarrow^* w\}.$$

A derivation of the form $S \Rightarrow^* w$ with $w \in T^*$ is called a terminal derivation.

## 3 Main Result

This section presents the main result concerning the descriptional complexity of semi-conditional grammars.

**Theorem 1** Every recursively enumerable language is generated by a semi-conditional grammar of degree $(2, 1)$ with no more than 7 conditional productions and 8 nonterminals.
PROOF. Let $L$ be a recursively enumerable language. There is a grammar $G' = (\{S, A, B, C\}, T, P \cup \{ABC \to \varepsilon\}, S)$ in the Geffert normal form such that $L = \mathcal{L}(G')$. Construct the grammar

$$G = (\{S, A, B, C, \#B', C'\}, T, P' \cup P'', S),$$

where

$$P' = \{(X \rightarrow \alpha, 0, 0) : X \rightarrow \alpha \in P\},$$

and $P''$ contains the following seven conditional productions:

1. $(A \rightarrow \#B', 0, \#B'),$
2. $(B \rightarrow B', \#, B'),$
3. $(C \rightarrow C'\#, \#B', C'),$
4. $(B' \rightarrow \varepsilon, B'C', 0),$
5. $(C' \rightarrow \varepsilon, \#C', 0),$
6. $(\# \rightarrow \varepsilon, \#B', 0),$
7. $(\$ \rightarrow \varepsilon, 0, \#).$

To prove that $L(G') \subseteq L(G)$, consider a derivation $S \Rightarrow^* wABCw'v \Rightarrow ww'v$ in $G'$ by productions from $P$ with only one application of the production $ABC \rightarrow \varepsilon$, where $w, w' \in \{A, B, C\}^*$ and $v \in T^*$. Then, $S \Rightarrow^* wABCw'v$ in $G$ by productions from $P'$. Moreover, by productions 1, 2, 3, 4, 5, 6, 7, 7, we get

$$wABCw'v \Rightarrow w\#BCw'v$$
$$\Rightarrow w\#B'Cw'v$$
$$\Rightarrow w\#B'C'w'v$$
$$\Rightarrow w\#C'w'v$$
$$\Rightarrow w\#w'v$$
$$\Rightarrow wv.$$ 

The inclusion follows by induction.

To prove that $L(G') \supseteq L(G)$, consider a terminal derivation. Let $X$ from $\{A, B, C\}$ be in a sentential form of this derivation. To eliminate $X$, there are the following three possibilities:

1. If $X = A$, then there must be $C$ and $B$ (by productions 6 and 3) in some (previous) sentential form;
2. If $X = B$, then there must be $C$ and $A$ (by productions 4 and 3) in some (previous) sentential form;
3. If $X = C$, then there must be $A$ and $B$ (by productions 5 and 3) in some (previous) sentential form.
In all above cases, there are $A$, $B$, and $C$ in some sentential form of the derivation. By productions 1, 2, 3, and 7, there cannot be more than one $#$, $B'$, and $C'$ in any sentential form. By productions 3 and 4, $#B'C'$ is a substring of a sentential form and there is no terminal symbol between any two nonterminals. Thus, the first part of any terminal derivation in $G$ is of the form

$$S \Rightarrow^* w_1ABCw_2w \Rightarrow^3 w_1$#B'C'$$w_2w$$

by productions from $P'$ and productions 1, 2, and 3, where $w_1 \in \{A, B\}^*$, $w_2 \in \{B, C\}^*$, and $w \in T^*$. Next, only production 4 is applicable. Thus,

$$w_1$#B'C'$w_2w \Rightarrow w_1$#C'$w_2w.$$

Besides a possible application of production 2, only production 5 is applicable. Thus,

$$w_1$#C'$w_2w \Rightarrow w_1$#C'$w_2w.$$ 

where $w'_1 \in \{A, B, B'\}^*$, $w'_2 \in \{B, B', C\}^*$. Besides a possible application of production 2, only production 6 is applicable. Thus,

$$w'_1$#C'$w'_2w \Rightarrow^+ w''_1$#C'$w''_2w$$

where $w''_1 \in \{A, B, B'\}^*$, $w''_2 \in \{B, B', C\}^*$. Finally, only production 7 is applicable, i.e.,

$$w''_1$#C'$w''_2w \Rightarrow^2 w''_1w''_2w.$$ 

Then,

$$w''_1w''_2w \Rightarrow^* uvw$$

by productions 1, 2, 3, or 1, 3, if production 2 has already been applied, where

$$uvw \in \{u_1$#B'C'$u_2w : u_1 \in \{A, B\}^*, u_2 \in \{B, C\}^*\}$$

or $uv = \varepsilon$. Thus, the substring $ABC$ and only this substring was eliminated. By induction (see (1)), the inclusion holds.
Acknowledgements

This work was supported by the Czech Grant Agency projects 201/07/0005 and 102/05/H050, FRVŠ grant FR762/2007/G1, and the Czech Ministry of Education under the Research Plan MSM 0021630528.

References


