A Note on the Generative Power of Some Simple Variants of Context-Free Grammars Regulated by Context Conditions

Tomáš Masopust

Brno University of Technology, Faculty of Information Technology
Božetěchova 2, Brno 61266, Czech Republic
masopust@fit.vutbr.cz

Abstract. This paper answers three open questions concerning the generative power of some simple variants of context-free grammars regulated by context conditions. Specifically, it discusses the generative power of so-called context-free semi-conditional grammars (which are random context grammars where permitting and forbidding sets are replaced with permitting and forbidding strings) where permitting and forbidding strings of each production are of length no more than one, and of simple semi-conditional grammars where, in addition, no production has attached both a permitting and a forbidding string. Finally, this paper also presents some normal form results, an overview of known results, and unsolved problems.

Key words: Formal languages; context condition; context-free grammar; random context grammar; semi-conditional grammar; simple semi-conditional grammar; erasing production; generative power.

1 Introduction

It is well-known that context-free languages play an important role in the theory and practice of formal languages in computer science. However, there is a lot of interesting and simple languages that are not context-free. According to the Chomsky hierarchy, such languages are treated as being context-sensitive. On the other hand, in the theory of regulated rewriting, many of these languages can be generated by regulated grammars using the benefits of applying only context-free productions.

The present paper discusses two simple variants of context-free grammars regulated by context conditions; both variants are special cases of so called *context-free random context grammars* (defined and studied by van der Walt in [1]), which are context-free grammars where two sets of symbols (*conditions*) are attached to each production—a permitting and a forbidding set. In addition, the grammars studied in this paper require that both the permitting and the forbidding sets contain no more than one symbol. A production of such a grammar is applicable to a sentential form provided that the symbol from the attached permitting set (the *permitting condition*) occurs in the sentential form while, simultaneously, the symbol from the attached forbidding set (the *forbidding condition*) does not. These grammars were defined by Păun [2] in 1985 and called *semi-conditional grammars of degree* $(1,1)$. In general, semi-conditional grammars are defined to be of any degree (i,j), for $i,j \geq 0$, where the degree (i,j) means that all permitting and forbidding conditions (that are strings, in general, not only symbols) are of length no more than i and j, respectively. In addition, semi-conditional grammars where each production has no more than one condition in the union of its permitting and forbidding sets are referred to as *simple semi-conditional grammars* (see [3]).

Since their introduction, it has been an open problem whether every semi-conditional grammar can be converted to an equivalent simple semi-conditional grammar (of the same degree); cf. [4, page 90]. This paper answers this question so that it demonstrates how to convert any semi-conditional grammar to an equivalent simple semi-conditional grammar of the same degree. In fact, this demonstration is given for both semi-conditional grammars with and without erasing productions. In addition, this paper also shows that semi-conditional grammars of degree $(1,1)$ characterize the family of recursively enumerable languages, which is a question left unsolved in [2] and still formulated as open in [4]. As an immediate consequence of these two results, it follows that simple semi-conditional grammars of degree $(1,1)$ characterize the family of recursively enumerable languages, too.

Furthermore, this paper presents three normal form results. Specifically, it proves that (i) for any (simple) semi-conditional grammar, there is an equivalent simple semi-conditional grammar of the same degree with the property that its (core context-free) productions can be decomposed into two disjoint sets in such a way that in one set, all productions have attached only permitting conditions, while in the other set, all productions have attached only forbidding conditions; it means that if $u_1, u_2, \ldots, u_k$ are all conditions attached to a production $A \to \alpha$, then all of them are either permitting or forbidding; (ii) for any context-sensitive language, there is a simple semi-conditional grammar of degree (i,j), $i,j \in \{1,2\}$, $i \neq j$, without erasing productions and without conditions containing terminal symbols that satisfies the property from (i); and (iii) for any recursively enumerable language, there is a simple semi-conditional grammar of degree $(1,1)$ without conditions containing terminal symbols that satisfies the property from (i).

In its conclusion, this paper gives an overview of known results concerning the generative power of discussed grammars, including the results concerning the descriptional complexity, presents a simple semi-conditional grammar of degree $(1,1)$ without erasing productions that generates a nontrivial context-sensitive language, and discusses open problems.

2 Preliminaries and Definitions

In this paper, we assume that the reader is familiar with formal language theory and with the theory of regulated rewriting (see [5, 6]). For an alphabet (finite nonempty set) V, V^* represents the free monoid generated by V. The unit of V^* is denoted by ε. Set $V^+ = V^* - \{\varepsilon\}$. For $w \in V^*$, $|w|$ denotes the length of w, and $sub(w) = \{u : u$ is a substring of $w\}$.

Let **RE**, **REC**, **CS**, **CF** denote the families of recursively enumerable, recursive, context-sensitive, and context-free languages, respectively. In addition, let $\mathbf{RC_{ac}}$, **RC**, and **fRC** denote the families of languages generated by random context grammars with appearance checking, random context grammars where each forbidding set is empty (*permitting grammars*), and random context grammars where each permitting set is empty (*forbidding grammars*), respectively. Moreover, superscript ε is added if erasing productions are allowed.

A *semi-conditional grammar* (see [2]) is a quadruple $G = (N, T, P, S)$, where N and T are the alphabets of nonterminals and terminals, respectively, such that $N \cap T = \emptyset$, $V = N \cup T$, $S \in N$ is the start symbol, and P is a finite set of productions of the form $(X \to \alpha, u, v)$ such that $X \to \alpha$ is a context-free production and $u, v \in V^+ \cup \{0\}$, where $0 \notin V$ is a special symbol. If for each production $(X \to \alpha, u, v) \in P$, $u \neq 0$ implies that $|u| \leq i$ and $v \neq 0$ implies that $|v| \leq j$, then G is said to be of *degree* (i, j). G is said to be *simple* if for each production $(X \to \alpha, u, v) \in P$ we have $0 \in \{u, v\}$.

For $x_1, x_2 \in V^*$, $x_1 X x_2 \Rightarrow x_1 \alpha x_2$ provided that

1. $(X \to \alpha, u, v) \in P$,
2. $u \neq 0$ implies that $u \in sub(x_1 X x_2)$, and
3. $v \neq 0$ implies that $v \notin sub(x_1 X x_2)$.

As usual, $\Rightarrow$ is extended to $\Rightarrow^i$, for $i \geq 0$, $\Rightarrow^+$, and $\Rightarrow^*$. The language of G is defined as $L(G) = \{w \in T^* : S \Rightarrow^* w\}$. The family of languages generated by semi-conditional grammars of degree (i, j) is denoted by $\mathbf{SC}^{\varepsilon}(\mathbf{i},\mathbf{j})$, or $\mathbf{SC}(\mathbf{i},\mathbf{j})$ if erasing productions are not allowed. Analogously, the family of languages generated by simple semi-conditional grammars of degree (i, j) is denoted by $\mathbf{SSC}^{\varepsilon}(\mathbf{i},\mathbf{j})$, or $\mathbf{SSC}(\mathbf{i},\mathbf{j})$ if erasing productions are not allowed.

3 Main Results

As stated above, this paper concentrates its attention on language families $\mathbf{SSC}^{\varepsilon}(\mathbf{1},\mathbf{1})$ and $\mathbf{SSC}(\mathbf{1},\mathbf{1})$. First, it answers three questions formulated as open in [4] (see also [2]) concerning the relations among the families $\mathbf{SSC}^{\varepsilon}(\mathbf{1},\mathbf{1})$, $\mathbf{SC}^{\varepsilon}(\mathbf{1},\mathbf{1})$, and **RE** (Theorems 3 and 2 and Corollary 1), and between the families $\mathbf{SSC}(\mathbf{1},\mathbf{1})$ and $\mathbf{SC}(\mathbf{1},\mathbf{1})$ (Theorem 1). Then, it gives an overview of known results, demonstrates the generative power of non-erasing simple semi-conditional grammars, and discusses open problems.

Theorem 1. *For any* $i, j \geq 1$, $\mathbf{SSC}(\mathbf{i},\mathbf{j}) = \mathbf{SC}(\mathbf{i},\mathbf{j})$.

Proof. Let $L \in \mathbf{SC}(\mathbf{i},\mathbf{j})$, for some $i, j \geq 1$. Then, there is a semi-conditional grammar $G = (N, T, P, S)$ of degree (i, j) without erasing productions such that $L(G) = L$. Construct a simple semi-conditional grammar $G' = (N', T, P', S_1)$, where S_1 is a new start symbol, $N' = N \cup \{S_1\} \cup \{[A] : A \in V\} \cup \{A', A'' : A \in N\} \cup \{[pA], [p_0 A], [p_1 A], [p_2 A], [p_3 A], [p' A], [p'' A] : p = (A \to \alpha, u, v) \in P\}$, $P' = \{(S_1 \to [S], 0, 0)\} \cup \{([a] \to a, 0, 0) : a \in T\}$, and for each $p = (A \to \alpha, u, v) \in P$, the following productions are added to P':

Case 1: For $B \in V$

1. $([B] \to [pB], u, 0)$,
2. $([B] \to [pB], [B]u', 0)$, for $u = Bu'$,
3. $([pB] \to [p_0 B], 0, v)$,

4. $([p_0B] \to [p_1B], 0, \gamma)$, where $\gamma = \begin{cases} [p_0B]v' & \text{if } v = Bv' \\ 0 & \text{otherwise} \end{cases}$
5. $(A \to A', [p_1B], 0)$,
6. $(A' \to A'', 0, A'')$,
7. $([p_1B] \to [p_2B], A'', 0)$,
8. $([p_2B] \to [p_3B], 0, A')$,
9. $(A'' \to \alpha, [p_3B], 0)$,
10. $([p_3B] \to [B], 0, A'')$.

Case 2: The first nonterminal of a sentential form is replaced.

11. $([A] \to [p'A], u, 0)$,
12. $([A] \to [p'A], [A]u', 0)$, for $u = Au'$,
13. $([p'A] \to [p''A], 0, v)$
14. $([p''A] \to [B]\beta, 0, \gamma)$, where $\alpha = B\beta$, $B \in V$, $\gamma = \begin{cases} [p''A]v' & \text{if } v = Av' \\ 0 & \text{otherwise.} \end{cases}$

To prove that $L(G) \subseteq L(G')$, consider a derivation of G. Such a derivation is of the form $S \Rightarrow^* Bw_1Aw_2 \Rightarrow Bw_1\alpha w_2$, where the last derivation step is made by a production $p = (A \to \alpha, u, v) \in P$, and $B \in V$. Then, G' derives as follows (numbers in square brackets denote (classes of) productions applied in given derivation steps):

$$\begin{aligned} S_1 &\Rightarrow [S] \\ &\Rightarrow^* [B]w_1Aw_2 \\ &\Rightarrow [pB]w_1Aw_2 \text{ [1 or 2]} \\ &\Rightarrow [p_0B]w_1Aw_2 \text{ [3]} \\ &\Rightarrow [p_1B]w_1Aw_2 \text{ [4]} \\ &\Rightarrow [p_1B]w_1A'w_2 \text{ [5]} \\ &\Rightarrow [p_1B]w_1A''w_2 \text{ [6]} \\ &\Rightarrow [p_2B]w_1A''w_2 \text{ [7]} \\ &\Rightarrow [p_3B]w_1A''w_2 \text{ [8]} \\ &\Rightarrow [p_3B]w_1\alpha w_2 \text{ [9]} \\ &\Rightarrow [B]w_1\alpha w_2 \text{ [10].} \end{aligned}$$

If the derivation is of the form $S \Rightarrow^* Aw \Rightarrow \alpha w = B\beta w$ in G, $B \in V$, i.e., the first nonterminal of the sentential form is replaced, then G' derives

$$\begin{aligned} S_1 &\Rightarrow [S] \\ &\Rightarrow^* [A]w \\ &\Rightarrow [p'A]w \text{ [11 or 12]} \\ &\Rightarrow [p''A]w \text{ [13]} \\ &\Rightarrow [B]\beta w \text{ [14].} \end{aligned}$$

The proof now proceeds by induction.

On the other hand, to prove that $L(G') \subseteq L(G)$, consider a sentential form $[A]w$ and assume that a production constructed in 11 or 12 is applied. Then, the only derivation is

$$\begin{aligned}[A]w &\Rightarrow [p'A]w \ [11 \text{ or } 12] \\ &\Rightarrow [p''A]w \ [13] \\ &\Rightarrow [B]\beta w \ [14],\end{aligned}$$

where $p = (A \to B\beta, u, v) \in P$, $B \in V$. From this, by a production constructed in 11 or 12, it follows that u (different from 0) is a substring of Aw, and, by productions constructed in 13 and 14, v is not a substring of Aw. Thus, $Aw \Rightarrow B\beta w$ by $(A \to B\beta, u, v) \in P$ in G.

Now, assume that a production constructed in 1 or 2 is applied to a sentential form $[B]w_1Aw_2$. Then, the only derivation is of the form

$$\begin{aligned}[B]w_1Aw_2 &\Rightarrow [pB]w_1Aw_2 \ [1 \text{ or } 2] \\ &\Rightarrow [p_0B]w_1Aw_2 \ [3] \\ &\Rightarrow [p_1B]w_1Aw_2 \ [4] \\ &\Rightarrow [p_1B]w_1A'w_2 \ [5] \\ &\Rightarrow [p_1B]w_1A''w_2 \ [6] \\ &\Rightarrow [p_2B]w_1A''w_2 \ [7] \\ &\Rightarrow [p_3B]w_1A''w_2 \ [8] \\ &\Rightarrow [p_3B]w_1\alpha w_2 \ [9] \\ &\Rightarrow [B]w_1\alpha w_2 \ [10],\end{aligned}$$

where $p = (A \to \alpha, u, v) \in P$. Surely, by a production constructed in 1 or 2, it follows that u (different from 0) is a substring of Bw_1Aw_2, and, by productions constructed in 3 and 4, it follows that v is not a substring of Bw_1Aw_2. Moreover, by a production constructed in 6, only one A' can be replaced with A'', and a production constructed in 8 can be applied only if there is no A'. Therefore, only one A is replaced with A' by a production constructed in 5 (the one later replaced with A''). Thus, only one A is replaced with α, i.e., $Bw_1Aw_2 \Rightarrow Bw_1\alpha w_2$ by $(A \to \alpha, u, v) \in P$ in G.

We have proved that $\mathbf{SC(i,j)} \subseteq \mathbf{SSC(i,j)}$. The other inclusion follows immediately from the definition. Hence, the theorem holds. □

Considering Case 2 of the previous construction, it is not hard to see that this construction is not valid for grammars with erasing productions; by an erasing production, the special first nonterminal of the form $[A]$ would be eliminated and the derivation would be blocked. However, a simple modification of the previous construction proves the following theorem.

Theorem 2. *For any $i, j \geq 1$, $\mathbf{SSC}^{\varepsilon}\mathbf{(i,j)} = \mathbf{SC}^{\varepsilon}\mathbf{(i,j)}$.*

Proof. Let $L \in \mathbf{SC}^{\varepsilon}\mathbf{(i,j)}$, for some $i, j \geq 1$. Then, there is a semi-conditional grammar $G = (N, T, P, S)$ of degree (i, j) such that $L(G) = L$. Construct a simple semi-conditional grammar $G' = (N', T, P', S_1)$, where $N' = N \cup \{S_1, X\} \cup \{A', A'' : A \in N\} \cup \{[p], [p_0], [p_1], [p_2] : p = (A \to \alpha, u, v) \in P\}$, S_1 and X are new symbols not in N, $P' = \{(S_1 \to XS, 0, 0), (X \to \varepsilon, 0, 0)\}$, and for each $p = (A \to \alpha, u, v) \in P$, the following productions are added to P':

1. $(X \to [p], u, 0)$,
2. $([p] \to [p_0], 0, v)$,
3. $(A \to A', [p_0], 0)$,
4. $(A' \to A'', 0, A'')$,
5. $([p_0] \to [p_1], A'', 0)$,
6. $([p_1] \to [p_2], 0, A')$,
7. $(A'' \to \alpha, [p_2], 0)$,
8. $([p_2] \to X, 0, A'')$.

The rest of the proof is analogous to the proof of Theorem 1 and is left to the reader. □

The following theorem answers the question left unsolved in [2] of what is the relation between the families $\mathbf{SC}^{\varepsilon}(\mathbf{1},\mathbf{1})$ and **RE**?

Theorem 3. $\mathbf{SC}^{\varepsilon}(\mathbf{1},\mathbf{1}) = \mathbf{RE}$.

Proof. The proof is a straightforward consequence of the proof given in [7, Section 3.2], where for each recursively enumerable language L, a random context grammar G is given such that $L(G) = L$ and each of permitting and forbidding sets contains no more than one symbol. The main idea of the proof is based on the fact that any recursively enumerable language can be generated by an unordered scattered context grammar. Then, such an unordered scattered context grammar in a special normal form generating L is consider and transformed into a random context grammar. For more details, the reader is referred to Lemma 6 in [7].

Thus, we obtain the required semi-conditional grammar by replacing one-element sets with their elements and empty sets with 0. □

As an immediate consequence, we have the following result.

Corollary 1. $\mathbf{SSC}^{\varepsilon}(\mathbf{1},\mathbf{1}) = \mathbf{RE}$.

As no context-free production in the constructions of Theorems 1 and 2 has attached both a permitting and a forbidding condition, the following corollary holds. It says that the core context-free productions can be decomposed into two disjoint sets of productions—the productions with only permitting conditions (*permitting productions*) and the productions with only forbidding conditions (*forbidding productions*). Note that in case of erasing productions, such systems have been studied (using a different technique) in [8] (cf. Corollary 4). Thus, the following consequences of the previous results of this paper complement [8] in case of non-erasing productions, and, in addition, use much simpler proofs than used in [8].

Corollary 2. *For any semi-conditional grammar G' of degree (i, j) without erasing productions, $i, j \geq 1$, there is an equivalent simple semi-conditional grammar $G = (N, T, P, S)$ of the same degree without erasing productions such that $(A \to \alpha, u, 0) \in P$ and $(A \to \alpha, 0, v) \in P$ imply that $0 \in \{u, v\}$.*

In addition, by a standard technique, it can be proved that conditions u and v contain only nonterminals, i.e., $u, v \in N^+ \cup \{0\}$, so that each production $(A \to \alpha, u, v)$ is replaced with $(A \to h(\alpha), h(u), h(v))$, where h is a homomorphism defined as $h(A) = A$, for

$A \in N \cup \{0\}$, and $h(a) = a'$, for $a \in T$, where a' is a new nonterminal, and $([a] \to a, 0, 0)$ is replaced with $([a] \to t_a, 0, 0)$ and $(t_a \to a, 0, 0)$, where t_a is a new nonterminal for all $a \in T$. Finally, $(a' \to a, t_b, 0)$, for $b \in T$, are added for all $a \in T$. In case of erasing productions, Theorem 2, $(X \to \varepsilon, 0, 0)$ is replaced with $(X \to Y, 0, 0)$ and $(Y \to \varepsilon, 0, 0)$, where Y is a new nonterminal, and $(a' \to a, Y, 0)$ are added for all $a \in T$.

By 4 of Theorem 5, we have the following normal form theorem.

Corollary 3. *For any context-sensitive language L, there is a simple semi-conditional grammar $G = (N, T, P, S)$ of degree (i, j), for $i, j \in \{1, 2\}$, $i \neq j$, without erasing productions such that $L(G) = L$ and*

1. $(A \to \alpha, u, v) \in P$ *implies that* $u, v \in N^+ \cup \{0\}$, *and*
2. $(A \to \alpha, u, 0) \in P$ *and* $(A \to \alpha, 0, v) \in P$ *imply that* $0 \in \{u, v\}$.

In addition, by Theorem 3, Corollary 3 can be modified to obtain the following normal form theorem.

Corollary 4. *For any recursively enumerable language L, there is a simple semi-conditional grammar $G = (N, T, P, S)$ such that $L(G) = L$ and*

1. $(A \to \alpha, u, v) \in P$ *implies that* $u, v \in N \cup \{0\}$ *(i.e., G is of degree $(1, 1)$), and*
2. $(A \to \alpha, u, 0) \in P$ *and* $(A \to \alpha, 0, v) \in P$ *imply that* $0 \in \{u, v\}$.

4 Overview of Results and Open Problems

This section presents an overview of results concerning simple semi-conditional grammars known so far. In addition, it also presents an overview of open problems.

Theorem 4. *The following holds for grammars with erasing productions.*

1. $\mathbf{SSC}^{\varepsilon}(\mathbf{0}, \mathbf{0}) = \mathbf{CF}$.
2. $\mathbf{CF} \subset \mathbf{SSC}^{\varepsilon}(\mathbf{0}, \mathbf{1}) \subseteq \mathbf{fRC}^{\varepsilon} \subset \mathbf{REC}$.
3. $\mathbf{CF} \subset \mathbf{SSC}^{\varepsilon}(\mathbf{1}, \mathbf{0}) \subseteq \mathbf{RC}^{\varepsilon} \subset \mathbf{REC}$.
4. $\mathbf{SSC}^{\varepsilon}(\mathbf{1}, \mathbf{1}) = \mathbf{SC}^{\varepsilon}(\mathbf{1}, \mathbf{1}) = \mathbf{RE}$.

Proof. The inclusions in 2 and 3 are straightforward; the proofs of the proper inclusions can be found, e.g., in [4] and [9], respectively. □

Theorem 5. *The following holds for grammars without erasing productions.*

1. $\mathbf{SSC}(\mathbf{0}, \mathbf{0}) = \mathbf{CF}$.
2. $\mathbf{CF} \subset \mathbf{SSC}(\mathbf{0}, \mathbf{1}) \subseteq \mathbf{fRC} \subset \mathbf{CS}$.
3. $\mathbf{CF} \subset \mathbf{SSC}(\mathbf{1}, \mathbf{0}) \subseteq \mathbf{RC} \subset \mathbf{CS}$.
4. $\mathbf{SSC}(\mathbf{2}, \mathbf{1}) = \mathbf{SSC}(\mathbf{1}, \mathbf{2}) = \mathbf{CS}$.
5. $\mathbf{SSC}(\mathbf{1}, \mathbf{1}) = \mathbf{SC}(\mathbf{1}, \mathbf{1}) \subseteq \mathbf{RC_{ac}} \subset \mathbf{CS}$.

Proof. The inclusions in 2 and 3 are straightforward; the proofs of the proper inclusions can be found, e.g., in [10] and [11], respectively. Results of 4 are proved in [4]. □

Note that the generative power of simple semi-conditional grammars of degree $(0,i)$ and $(i,0)$ (with or without erasing productions), for $i \geq 2$, are not known. However, if more than one forbidding string is allowed to be attached to a production (i.e., there are sets of forbidding conditions instead of only one condition), it is known that such grammars (referred to as *generalized forbidding grammars*) are computationally complete. In addition, it is sufficient to have no more than four forbidding conditions each of which is of length no more than two to characterize the family of recursively enumerable languages (see [12, Corollary 6]). On the other hand, however, the question of what is the generative power of *generalized permitting grammars* (defined in the same manner) is an open problem.

Note also that the precise relation between $\mathbf{SSC(1,1)}$ and $\mathbf{RC_{ac}}$ is not known. However, the following theorem illustrates the generative power of simple semi-conditional grammars so that it shows that they are powerful enough to generate nontrivial languages, such as prime numbers, i.e., the language $\mathbf{P} = \{a^p : p \text{ is a prime number}\}$.

Theorem 6. $\mathbf{P} \in \mathbf{SSC(1,1)}$.

Proof. Let $G = (N, \{a\}, P, S')$ be a simple semi-conditional grammars, where N follows from P that is constructed as follows:

1. $(S' \to a^2, 0, 0)$
2. $(S' \to S, 0, 0)$
3. $(S \to SCC, 0, 0)$
4. $(S \to AAX, 0, 0)$
5. $(A \to \bar{A}, X, 0)$
6. $(C \to \bar{C}, X, 0)$
7. $(\bar{A} \to A', 0, A')$
8. $(\bar{C} \to C', 0, C')$
9. $(X \to Z_1, A', 0)$
10. $(X \to Y_1, 0, A)$
11. $(X \to Q_1, 0, C)$
12. $(X \to F, 0, C)$
13. $(Z_1 \to Z_2, C', 0)$
14. $(Z_2 \to Z_3, 0, \bar{A})$
15. $(Z_3 \to Z, 0, \bar{C})$
16. $(A' \to B, Z, 0)$
17. $(C' \to D, Z, 0)$
18. $(Z \to Z_4, 0, A')$
19. $(Z_4 \to X, 0, C')$
20. $(Y_1 \to Y_2, 0, \bar{A})$
21. $(Y_2 \to Y, 0, A')$
22. $(B \to A, Y, 0)$
23. $(Y \to X, 0, B)$
24. $(Q_1 \to Q_2, 0, \bar{C})$
25. $(Q_2 \to Q_3, 0, C')$
26. $(Q_3 \to Q_4, 0, \bar{A})$
27. $(Q_4 \to Q_5, A', 0)$
28. $(Q_5 \to Q, A, 0)$
29. $(D \to \bar{D}, Q, 0)$
30. $(D \to C, Q, 0)$
31. $(B \to A, Q, 0)$
32. $(A' \to A, Q, 0)$
33. $(\bar{D} \to D_1, 0, D_1)$
34. $(Q \to Q_6, 0, D)$
35. $(Q_6 \to Q_7, 0, \bar{D})$
36. $(Q_7 \to Q_8, D_1, 0)$
37. $(D_1 \to A, Q_8, 0)$
38. $(Q_8 \to Q_9, 0, D_1)$
39. $(Q_9 \to Q_{10}, 0, B)$
40. $(Q_{10} \to X, 0, A')$
41. $(A \to a, F, 0)$
42. $(F \to a, 0, A)$

We prove that $L(G) = \mathbf{P}$. Clearly, a^2 is in $\mathbf{P}$. Thus, consider a terminal derivation beginning by an application of production 2. Then, only productions 3 and 4 are applicable, generating the sentential form $AAX(CC)^n$, for some $n \geq 0$, i.e., from now on, any sentential form is of length $2k+1$, for some $k \geq 1$.

Now, only productions 5, 6, 9, 10, 11, and 12 are applicable; of course, if productions 5 and 6 are applicable, then they are applied before any of productions 9, 10, 11, or 12.

A. Let production 9 be applied. Then, clearly, productions 7 and 8 had to be applied before productions 13 and 9, respectively. Then, by productions 13 to 19, the derivation continues according to these productions as follows:

$$A^q B^m C^r D^m X \Rightarrow^* A^{q-1} B^{m+1} C^{r-1} D^{m+1} X\,.$$

(Note that symbols of sentential forms are written in the alphabetic order, rather than in the actual possible order, because the order is not important.) Informally, this phase of the derivation replaces one A with B and one C with D, respectively.

B. Let production 10 be applied. Then, by productions 20 to 23, the derivation replaces each B with A, i.e.,

$$B^n C^r D^{tn} X \Rightarrow^* A^n C^r D^{tn} X\,.$$

Together with the previous phase, these two phases try to divide $2k+1$ by n, where $n \geq 2$.

C. Let production 11 be applied. Then, by productions 24 to 40, the derivation continues so that it verifies that there is no C (including C' and $\bar{C}$) and $\bar{A}$ and that there is A' and A in the current sentential form. Then, precisely one D_1 is generated from D, and each other D is replaced with C. Finally, it verifies that all symbols B and A' are replaced with A. Thus, we have

$$A^{n-m} B^m D^{tn+m} X \Rightarrow^* A^{n+1} C^{tn+m-1} X\,.$$

This phase verifies that n does not divide $2k+1$ so that it requires the reminder to be at least one (symbols A and A' are required to be in the sentential form; one of them is compared against the symbol X, the other is the nonzero remainder). More precisely, if there were $m \geq 2$ such that $2k+1 = mn$, then

$$A^n C^{n-1} D^{(m-2)n} X \Rightarrow^* A' B^{n-1} D^{n-1} D^{(m-2)n} Q_5$$

and the derivation would be blocked (see production 28).

D. Let production 12 be applied. Then, by productions 41 and 42, the derivation continues according to these productions as follows:

$$A^{2k} X \Rightarrow A^{2k} F \Rightarrow^* a^{2k+1}\,,$$

where $2k+1$ is a prime number because the derivation has verified that there is no $n \in \{2, 3, \ldots, 2k-1\}$ such that n divides $2k+1$.

Thus, the whole derivation is of the form

$$\begin{aligned} A^2 C^{2(k-1)} X &\Rightarrow^* B^2 C^{2(k-2)} D^2 X \\ &\Rightarrow^* B^2 D^{2(k-1)} X \\ &\Rightarrow^* A^2 D^{2(k-1)} X \\ &\Rightarrow^* A^3 C^{2(k-1)-1} X \\ &\Rightarrow^* A^4 C^{2(k-2)} X \\ &\Rightarrow^* A^{2k} X \\ &\Rightarrow^* a^{2k+1}\,, \end{aligned}$$

where $2k+1$ is a prime number, i.e., $L(G) = \{a^p : p \text{ is a prime number}\} = \mathbf{P}$. □

5 Conclusion

From both theoretical and practical points of view, it is of a great interest to know the amount of resources needed to characterize any recursively enumerable language by (simple) semi-conditional grammars. This section summarizes results concerning the descriptional complexity of (simple) semi-conditional grammars known so far.

Let $(A \rightarrow \alpha, u, v)$ be a production of a semi-conditional grammar. If $u = v = 0$, then the production is said to be context-free; otherwise, it is said to be *conditional*.

Theorem 7 ([13]). *Every recursively enumerable language is generated by a simple semi-conditional grammar of degree* $(3,1)$ *with no more than eight conditional productions and eleven nonterminals.*

Theorem 8 ([14]). *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.*

In case of semi-conditional grammars that are not simple, the previous result can be improved as follows.

Theorem 9 ([15]). *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.*

Finally, note that Example 4.1.1 in [5] shows that there is no bound for the number of nonterminals for (simple) semi-conditional grammars of degree $(1,1)$ if terminal symbols are not allowed to appear in the conditions. More specifically, the example shows that any (simple) semi-conditional grammar of degree $(1,1)$ generating the language $T_n = \bigcup_{i=1}^{n} \{a_i^j : j \geq 1\}$, where conditions are nonterminal symbols, requires, in the nonerasing case, exactly $n+1$ nonterminal symbols, and, in the erasing case, at least $f(n)$ nonterminal symbols, for some unbounded mapping $f : \mathbb{N} \rightarrow \mathbb{N}$. In general, however, as terminal symbols are allowed to appear in the conditions, and $G = (\{S,A\}, \{a_1, a_2, \ldots, a_n\}, \{(S \rightarrow a_iA, 0, 0), (S \rightarrow a_i, 0, 0), (A \rightarrow a_iA, a_i, 0), (A \rightarrow a_i, a_i, 0) : 1 \leq i \leq n\}, S)$ is a simple semi-conditional grammar of degree $(1,1)$ that generates T_n, the question of whether analogous descriptional complexity results can be achieved for general (simple) semi-conditional grammars of degree $(1,1)$ is open. Furthermore, other cases not presented above are open, too.

To summarize the main results, this paper has answered three questions formulated as open in [4, page 90] (see also [2], where semi-conditional grammars were introduced and studied). Specifically, it has proved that

1. every semi-conditional grammar (with or without erasing productions) can be converted to an equivalent simple semi-conditional grammar (with or without erasing productions, respectively) of the same degree,
2. semi-conditional grammars of degree $(1,1)$ characterize the family of recursively enumerable languages,
3. and, as a consequence, simple semi-conditional grammars of degree $(1,1)$ characterize the family of recursively enumerable languages.

In addition, it has also presented some normal form results and an overview of known results, demonstrated the generative power of simple semi-conditional grammars of degree $(1, 1)$ without erasing productions, and discussed open problems.

Acknowledgments

The author thanks the anonymous referees for their helpful suggestions.

This work was supported by the Czech Ministry of Education under the Research Plan No. MSM 0021630528 and, partially, by the Czech Grant Agency project No. 201/07/0005.

References

1. van der Walt, A.P.J.: Random context grammars. In: Proceedings of the Symposium on Formal Languages. (1970)
2. Păun, G.: A variant of random context grammars: Semi-conditional grammars. Theoretical Computer Science **41** (1985) 1–17
3. Gopalaratnam, A., Meduna, A.: On semi-conditional grammars with productions having either forbidding or permitting conditions. Acta Cybernetica **11**(4) (1994) 307–324
4. Meduna, A., Švec, M.: Grammars with Context Conditions and Their Applications. John Wiley & Sons, New York (2005)
5. Dassow, J., Păun, G.: Regulated Rewriting in Formal Language Theory. Springer-Verlag, Berlin (1989)
6. Salomaa, A.: Formal languages. Academic Press, New York (1973)
7. Mayer, O.: Some restrictive devices for context-free grammars. Information and Control **20** (1972) 69–92
8. Masoput, T., Meduna, A.: On context-free rewriting with a simple restriction and its computational completeness. To appear in RAIRO-ITA
9. Bordihn, H., Fernau, H.: Accepting grammars and systems. Technical Report 9/94 (1994)
10. van der Walt, A.P.J., Ewert, S.: A shrinking lemma for random forbidding context languages. Theoretical Computer Science **237**(1-2) (2000) 149–158
11. Ewert, S., van der Walt, A.P.J.: A pumping lemma for random permitting context languages. Theoretical Computer Science **270**(1–2) (2002) 959–967
12. Masopust, T., Meduna, A.: Descriptional complexity of generalized forbidding grammars. In: Proceedings of 9th International Workshop on Descriptional Complexity of Formal Systems, High Tatras, Slovakia (2007) 170–177
13. Vaszil, G.: On the descriptional complexity of some rewriting mechanisms regulated by context conditions. Theoretical Computer Science **330** (2005) 361–373
14. Masopust, T., Meduna, A.: Descriptional complexity of grammars regulated by context conditions. In: Pre-proceedings of 1st International Conference on Language and Automata Theory and Application (LATA 2007), Tarragona, Spain (2007) 403–411
15. Masopust, T., Meduna, A.: Descriptional complexity of semi-conditional grammars. Information Processing Letters **104**(1) (2007) 29–31