DCFS (Draft)
2009

Descriptional Complexity of Three-Nonterminal Scattered Context Grammars: An Improvement

Tomáš Masopust   Alexander Meduna

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

Abstract. Recently, it has been shown that every recursively enumerable language can be generated by a scattered context grammar with no more than three nonterminals. However, in that construction, the maximal number of nonterminals simultaneously rewritten during a derivation step depends on many factors, such as the cardinality of the alphabet of the generated language and the structure of the generated language itself. This paper improves the result by showing that the maximal number of nonterminals simultaneously rewritten during any derivation step can be limited by a small constant regardless of other factors.

Keywords: scattered context grammars, descriptional complexity, generative power

1   Introduction

Scattered context grammars, introduced by Greibach and Hopcroft in [2], are partially parallel rewriting devices based on context-free productions, where in each derivation step, a finite number of nonterminal symbols of the current sentential form is simultaneously rewritten. As scattered context grammars were originally defined without erasing productions, it is no surprise that they generate only context sensitive languages. On the other hand, however, the question of whether every context sensitive language can be generated by a (nonerasing) scattered context grammar is an interesting, longstanding open problem. Note that the natural generalization of these grammars allowing erasing productions makes them computationally complete (see [6]). For some conditions when a scattered context grammar can be transformed to an equivalent nonerasing scattered context grammar, the reader is referred to [9]. In what follows, we implicitly consider scattered context grammars with erasing productions.

Although many interesting results have been achieved in the area of the descriptional complexity of scattered context grammars during the last few decades, the main motivation to re-open this investigation area comes from an interesting, recently started research project on bulding parsers and compilers of programming languages making use of advantages of scattered context grammars (see, for instance, papers [3, 10] for more information on the advantages and problems arising from this approach).

To give an insight into the descriptional complexity of scattered context grammars (including erasing productions), note that it is proved in [8] that one-nonterminal scattered context grammars are not powerful enough to generate all context sensitive languages so that it is demonsrated that they are not able to generate the language $\{a^{2^{2^n}} : n \geq 0\}$ (which is scattered context, see Lemma 2 below). In addition, although they are not able to generate all these languages, it is an open problem (because of the erasing productions) whether they can generate a language which is not context sensitive. On the other hand, it is proved in [7] that three nonterminals are sufficient enough for scattered context grammars to characterize the family of recursively enumerable languages. In that proof, however, the maximal number of nonterminal symbols simultaneously rewritten during any derivation step depends on the alphabet of the generated language and on the structure of the generated language itself.

Later, in [12], Vaszil gave another construction limiting the maximal number of nonterminals simultaneously rewritten during one derivation step. However, this improvement is for the price of increasing the number of nonterminals. Although Vaszil's construction has been improved since then (in the sense of the number of nonterminals, see [5] for an overview of the latest results), the number of three nonterminals has not been achieved.

This paper presents a construction improving the descriptional complexity of scattered context grammars with three nonterminals by limiting the maximal number of nonterminals simultaneously rewritten during any derivation step regardless of any other factors. This result is achieved by the combination of approaches of both previously mentioned papers. Specifically, this paper proves that every recursively enumerable language is generated by a three-nonterminal scattered context grammar, where no more than nine symbols are simultaneously rewritten during any derivation step. This is a significant improvement in comparison with the result of [7], where more than $2n + 4$ symbols have to be simultaneously rewritten during almost all derivation steps of any successful derivation, for some n strictly greater than the number of terminal symbols of the generated language plus two. To be more precise, n strongly depends not only on the terminal alphabet of the generated language, but also on the structure of the generated language itself.

Finally, note that analogously as in [7], we do not give a constant limit on the number of non-context-free productions, which is also limited by fixed constants in [12] and [5]. To find such a limit is an interesting challenge for the future research, as well as to find out whether the number of nonterminals can be reduced to two. See also the overview of known results and open problems in the conclusion.

2   Preliminaries and Definitions

We assume that the reader is familiar with formal language theory (see [11]). For an alphabet (finite nonempty set) V, V^* represents the free monoid generated by

V with the unit denoted by λ. Set $V^+ = V^* - \{\lambda\}$. For $w \in V^*$ and $a \in V$, let $|w|$, $|w|_a$, and w^R denote the length of w, the number of occurrences of a in w, and the mirror image of w, respectively.

A *scattered context grammar* is a quadruple $G = (N, T, P, S)$, where N is the alphabet of nonterminals, T is the alphabet of terminals such that $N \cap T = \emptyset$, $S \in N$ is the start symbol, and P is a finite set of productions of the form $(A_1, A_2, \ldots, A_n) \to (x_1, x_2, \ldots, x_n)$, for some $n \geq 1$, where $A_i \in N$ and $x_i \in (N \cup T)^*$, for all $i = 1, \ldots, n$. If $n \geq 2$, then the production is said to be *non-context-free*; otherwise, it is context-free. In addition, if for each $i = 1, \ldots, n$, $x_i \neq \lambda$, then the production is said to be *nonerasing*; G is *nonerasing* if all its productions are nonerasing.

For $u, v \in (N \cup T)^*$, $u \Rightarrow v$ in G provided that

- $u = u_1 A_1 u_2 A_2 u_3 \ldots u_n A_n u_{n+1}$,
- $v = u_1 x_1 u_2 x_2 u_3 \ldots u_n x_n u_{n+1}$, and
- $(A_1, A_2, \ldots, A_n) \to (x_1, x_2, \ldots, x_n) \in P$,

where $u_i \in (N \cup T)^*$, for all $i = 1, \ldots, n + 1$. The language generated by G is defined as $L(G) = \{w \in T^* : S \Rightarrow^* w\}$, where $\Rightarrow^*$ denotes the reflexive and transitive closure of the relation $\Rightarrow$. A language L is said to be a (nonerasing) *scattered context language* if there is a (nonerasing) scattered context grammar G such that $L = L(G)$.

3 Main Results

First, we give a simple example of a nonerasing scattered context grammar generating a non-context-free language. Then, we present a nonerasing scattered context grammar generating the nontrivial context sensitive language $\{a^{l^{k^n}} : n \geq 0\}$, for any $k, l \geq 2$. Thus, for $k = l = 2$, we have a scattered context grammar generating the language mentioned in the introduction. Note that independently on k and l, the grammar has only twelve nonterminals and fourteen productions, ten of which are non-context-free.

Example 1. Let $G = (\{S, A, B, C\}, \{a, b, c\}, P, S)$ be a scattered context grammar with P containing the following productions

- $(S) \to (ABC)$
- $(A, B, C) \to (aA, bB, cC)$
- $(A, B, C) \to (a, b, c)$

Then, it is not hard to see that the language generated by G is $L(G) = \{a^n b^n c^n : n \geq 1\}$.

Lemma 2. *For any $k, l \geq 2$, the language $\{a^{l^{k^n}} : n \geq 0\}$ is a nonerasing scattered context language.*

Proof: Let $G = (\{S, A, A', A'', B, C, X, X_2, X_3, Y, Z, Z'\}, \{a\}, P, S)$ be a nonerasing scattered context grammar with P containing the following productions:

(I) $(S) \to (a^l)$,

(II) $(S) \to (a^{l^k})$,

(III) $(S) \to \left(a^{l^{k^2}}\right)$,

(IV) $(S) \to (A''A^{l-1}X_2B^{k^2-3}A'C^{k^2-1}XY)$,

   *   first stage

(V) $(A',C,X,Y) \to (B^{k-1},A',X,C^kY)$,

(VI) $(A',X,Y) \to (B^{k-1},A',C^{k-1}XY)$,

(VII) $(A',X,Y) \to (Z,Z,Y)$,

(VIII) $(Z,C,Z,Y) \to (Z,B^{k-1},Z,Y)$,

(IX) $(Z,Z,Y) \to (B,B^{k-1},X_3)$,

   *   second stage

(X) $(A'',A,X_2,X_3) \to (a^{l-1},A'',X_2A^l,X_3)$,

(XI) $(A'',X_2,B,X_3) \to (a^{l-1},A'',A^{l-1}X_2,X_3)$,

(XII) $(A'',X_2,X_3) \to (Z',Z',X_3)$,

(XIII) $(Z',A,Z',X_3) \to (Z',a^{l-1},Z',X_3)$,

(XIV) $(Z',Z',X_3) \to (a,a^{l-1},a^{l-1})$.

Then, all the possible successful derivations of G are summarized in the following (strings in the square brackets are regular expressions describing the productions applied during the derivations).

$$
\begin{aligned}
S &\Rightarrow a^l & &[(\text{I})] \\
S &\Rightarrow a^{l^k} & &[(\text{II})] \\
S &\Rightarrow a^{l^{k^2}} & &[(\text{III})] \\
S &\Rightarrow A''A^{l-1}X_2B^{k^2-3}A'C^{k^2-1}XY & &[(\text{IV})] \\
&\Rightarrow^* A''A^{l-1}X_2B^{k^n-2}X_3 & &[((\text{V})^+(\text{VI}))^*(\text{VII})(\text{VIII})^+(\text{IX})] \\
&\Rightarrow^* a^{l^{k^n-1}-l}A''A^{l^{k^n-1}-1}X_2X_3 & &[((\text{X})^+(\text{XI}))^*] \\
&\Rightarrow^* a^{l^{k^n}} & &[(\text{XII})(\text{XIII})^+(\text{XIV})].
\end{aligned}
$$

The first three cases are clear. In the last case, l symbols A (including A'') are generated in the first derivation step. Then, the derivation can be divided into two parts: in the first part, only productions from the first stage are applied (because there is no X_3 in the sentential form) generating k^n auxiliary symbols (B's, X_2, and X_3). Then, in the second part, only productions from the second stage are applied (because there is no Y in the sentential form) generating l^{k^n} symbols a. More precisely, we prove that all sentential forms of a successful derivation containing X_3, i.e. of the second part, are of the form

$$
a^{l^{m-1}-l}A''A^{l^{m-1}-1}X_2B^{k^n-m}X_3,
$$

for all $m = 2, 3, \ldots, k^n$ and $n \geq 3$. Clearly, for $m = 2$, the sentential form is $A''A^{l-1}X_2B^{k^n-2}X_3$. For $m = k^n$, we have $a^{l^{k^n-1}-l}A''A^{l^{k^n-1}-1}X_2X_3$ and it is not

hard to prove that

$$a^{l^{k^n-1}-l}A''A^{l^{k^n-1}-1}X_2X_3 \Rightarrow^* a^{l^{k^n-1}-l}aa^{(l-1)(l^{k^n-1}-1)}a^{l-1}a^{l-1} = a^{l^{k^n}}.$$

Thus, assume that $2 \leq m < k^n$. Then,

$$
\begin{aligned}
&a^{l^{m-1}-l}A''A^{l^{m-1}-1}X_2B^{k^n-m}X_3 \\
\Rightarrow^* \ &a^{l^{m-1}-l}a^{(l-1)(l^{m-1}-1)}A''X_2A^{l(l^{m-1}-1)}B^{k^n-m}X_3 \quad [\text{(x)}^*]\\
= \ &a^{l^m-2l+1}A''X_2A^{l^m-l}B^{k^n-m}X_3 \\
\Rightarrow \ &a^{l^m-2l+1}a^{l-1}A''A^{l^m-l}A^{l-1}X_2B^{k^n-m-1}X_3 \quad [\text{(xi)}]\\
= \ &a^{l^m-l}A''A^{l^m-1}X_2B^{k^n-(m+1)}X_3.
\end{aligned}
$$

For a complete proof of the correctness of this construction, the reader is referred to [4]. □

Now, we prove the main result of this paper.

Theorem 3. *Every recursively enumerable language is generated by a scattered context grammar with three nonterminals, where no more than nine nonterminals are simultaneously rewritten during one derivation step.*

Proof: Let L be a recursively enumerable language. Then, by Geffert [1], there is a grammar $G' = (\{S', A, B, C, D\}, T, P \cup \{AB \to \lambda, CD \to \lambda\}, S')$, where P contains only context-free productions of the following three forms: $S' \to uS'a$, $S' \to uS'v$, $S' \to \lambda$, for $u \in \{A, C\}^*$, $v \in \{B, D\}^*$, and $a \in T$. In addition, Geffert proved that any successful derivation of G' is divided into two parts: the first part is of the form

$$S' \Rightarrow^* w_1 S' w_2 w \Rightarrow w_1 w_2 w,$$

generated only by context-free productions from P, where $w_1 \in \{A, C\}^*$, $w_2 \in \{B, D\}^*$, and $w \in T^*$, and the second part is of the form

$$w_1 w_2 w \Rightarrow^* w,$$

generated only by productions $AB \to \lambda$ and $CD \to \lambda$.

Let $G = (\{S, A, B\}, T, P, S)$ be a scattered context grammar with P constructed as follows:

(i) $(S) \to (SBBASABBSA)$
(ii) $(S, S, S) \to (S, h(u)Sh(a), S)$ if $S' \to uS'a \in P'$,
(iii) $(S, S, S) \to (S, h(u)Sh(v), S)$ if $S' \to uS'v \in P'$,
(iv) $(S, A, B, B, S, B, B, A, S) \to (\lambda, \lambda, \lambda, S, S, S, \lambda, \lambda, \lambda)$,
(v) $(S, B, A, B, S, B, A, B, S) \to (\lambda, \lambda, \lambda, S, S, S, \lambda, \lambda, \lambda)$,
(vi) $(S, B, B, A, S, A, B, B, S) \to (\lambda, \lambda, \lambda, SBBA, S, S, \lambda, \lambda, \lambda)$,
(vii) $(S, B, B, A, S, A, B, B, S) \to (\lambda, \lambda, \lambda, S, S, S, \lambda, \lambda, \lambda)$,
(viii) $(S, S, S, A) \to (\lambda, \lambda, \lambda, \lambda)$,

where h is a homomorphism from $(\{A, B, C, D\} \cup T)^*$ to $(\{A, B\} \cup T)^*$ defined as $h(A) = ABB$, $h(B) = BBA$, $h(C) = h(D) = BAB$, and $h(a) = AaBB$, for all $a \in T$.

To prove that $L(G') \subseteq L(G)$, consider a successful derivation of $w \in T^*$ in G'. Such a derivation is of the form described above, where the second part of the derivation is according to a sequence $p_1 p_2 \ldots p_r$ of productions $AB \to \lambda$ and $CD \to \lambda$, for some $r \geq 0$. Then, in G, the derivation of w can be simulated by applications of the corresponding productions constructed above as follows:

$$
\begin{aligned}
S &\Rightarrow SBBASABBSA \quad [(\text{I})] \\
&\Rightarrow^* SBBAh(w_1)Sh(w_2 w)ABBSA \quad [(\text{II})^*(\text{III})^*] \\
&\Rightarrow^* Sh(w_1)Sh(w_2)SwA \quad [(\text{VI})^*(\text{VII})] \\
&\Rightarrow^* SSSwA \quad [q_r \ldots q_2 q_1] \\
&\Rightarrow w \quad [(\text{VIII})],
\end{aligned}
$$

where $q_i = (S, A, B, B, S, B, B, A, S) \to (\lambda, \lambda, \lambda, S, S, S, \lambda, \lambda, \lambda)$ if $p_i = AB \to \lambda$, or $q_i = (S, B, A, B, S, B, A, B, S) \to (\lambda, \lambda, \lambda, S, S, S, \lambda, \lambda, \lambda)$ otherwise, for each $1 \leq i \leq r$.

On the other hand, to prove that $L(G) \subseteq L(G')$, we demonsrate that G' generates any $x \in L(G)$.

First, we prove that each of the productions (I) and (VIII) is applied exactly once in each successful derivation of G. To prove this, let $S \Rightarrow^* x$ be a derivation of a string $x \in (\{S, A, B\} \cup T)^*$. Let i be the number of applications of production (I), j be the number of applications of production (VIII), and $2k$ be the number of B's in x. Then, it is not hard to see that

- $|x|_B = 2k$,
- $|x|_A = k + i - j$,
- $|x|_S = 1 + 2i - 3j$.

Thus, for $x \in T^*$, we have that $2k = 0$ and $i = j$. In addition, $1 + 2i - 3i = 0$ implies that $i = 1$, which means that each of the productions (I) and (VIII) is applied exactly once in each successful derivation of G—production (I) as the first production and production (VIII) as the last production of the derivation. We have shown that every successful derivation of G is of the form

$$S \Rightarrow SBBASABBSA \Rightarrow^* w_1 S w_2 S w_3 S w_4 A \Rightarrow w_1 w_2 w_3 w_4,$$

for some terminal strings $w_1, w_2, w_3, w_4 \in T^*$.

Furthermore, there is no production that can change the position of the middle symbol S. Therefore, with respect to productions of G, we have that $w_1, w_2 \in \{A, B\}^*$, which along with $w_1, w_2 \in T^*$ implies that $w_1 = w_2 = \lambda$. Thus, the previously shown successful derivation is of the form

$$S \Rightarrow SBBASABBSA \Rightarrow^* SS w_3 S w_4 A \Rightarrow w_3 w_4.$$

Analogously, it can be seen that $w_3 \in \{BAB, BBA, AaBB : a \in T\}^*$. Therefore, from the same reason as above, $w_3 = \lambda$ and every successful derivation of G is of

the form

$$S \Rightarrow SBBASABBSA \Rightarrow^* SSSwA \Rightarrow w\,, \tag{1}$$

for some $w \in T^*$.

Consider any inner sentential form of a successful derivation of G. Such a sentential form is a string of the form

$$u_1 S u_2 S u_3 S u_4 A\,,$$

for some $u_i \in (\{A, B\} \cup T)^*$, $1 \leq i \leq 4$. However, it is not hard to see that $u_1 = \lambda$ and $u_4 \in T^*$; otherwise, if there is a nonterminal symbol appearing in the string $u_1 u_4$, then, according to the form of productions, none of these symbols can be removed and, therefore, the derivation cannot be successful. Thus, every inner sentential form of any successful derivation of G is of the form

$$S \bar{u} S \bar{v} S \bar{w} A\,, \tag{2}$$

where $\bar{u} \in (BBA + \lambda)\{ABB, BAB\}^*$, $\bar{v} \in \{BAB, BBA, AaBB : a \in T\}^*$, and $\bar{w} \in T^*$. Now, we prove that

$$\bar{v} \in \{BBA, BAB\}^*\{AaBB : a \in T\}^*(ABB + \lambda)\,.$$

In other words, we prove that any applications of productions (VI) and (VII) precede the first application of any of productions (IV) and (V).

Thus, consider the beginning of a successful derivation of the form

$$S \Rightarrow SBBASABBSA \Rightarrow^* SBBAuSvABBSwA\,,$$

where none of productions (VI) and (VII) has been applied, and the first application of one of these productions follows. Note that during this derivation, only productions (I) to (III) have been applied because the application of production (IV) or (V) skips some nonterminal symbols and, therefore, leads to an incorrect sentential form (see the correct form (2) above). Clearly, $w = \lambda \in T^*$ (it is presented here for the reason of induction).

If production (VI) follows, the derivation proceeds

$$SBBAuSvABBSwA \Rightarrow SBBAuSvSwA\,, \tag{3}$$

and if production (VII) follows, the derivation proceeds

$$SBBAuSvABBSwA \Rightarrow SuSvSwA\,. \tag{4}$$

In addition, $w \in T^*$ and, according to the form of productions (I) to (III), $u \in \{ABB, BAB\}^*$ and $v \in \{BBA, BAB, AaBB : a \in T\}^*$.

Now, productions (II) and (III) can be applied. Let

$$SBBAuSvSwA \Rightarrow^* SBBAuu_1 Sv_1 vSwA \quad [((\text{II})+(\text{III}))^*] \tag{5}$$

and

$$SuSvSwA \Rightarrow^* Suu_1 Sv_1 vSwA \quad [((\text{II})+(\text{III}))^*] \tag{6}$$

be the longest parts of the derivation by productions (II) and (III), i.e., the application of one of productions (IV) to (VIII) follows.

I. In the first case, derivation (5), each of productions (IV), (V), and (VIII) leads to an incorrect sentential form. Thus, either production (VI) or (VII) has to be applied. In both cases, however, v_1v has to be of the form $v'AaBB$, for some $a \in T$, i.e.,

$$SBBAuu_1Sv'AaBBSwA \Rightarrow SBBAu'Sv'SawA \quad [(\text{VI})] \tag{7}$$

and the derivation proceeds as in (5) or

$$SBBAuu_1Sv'AaBBSwA \Rightarrow Su'Sv'SawA \quad [(\text{VII})] \tag{8}$$

and the derivation proceeds as in (6) because $u' = uu_1 \in \{ABB, BAB\}^*$ and $v' \in \{BBA, BAB, AaBB : a \in T\}^*$. By induction,

$$SBBAu'Sv'SawA \Rightarrow^* Su''Sv''Sw''awA \quad [((\text{II})+(\text{III})+(\text{VI}))^*(\text{VII})], \tag{9}$$

for some $u'' \in \{ABB, BAB\}^*$, $v'' \in \{BBA, BAB, AaBB : a \in T\}^*$, and $w''aw \in T^*$.

II. In the second case, derivation (6), each of productions (VI) and (VII) leads to an incorrect sentential form, and production (VIII) finishes the derivation, which, as shown above, implies that $uu_1 = v_1v = \lambda$. Thus, assume that either production (IV) or production (V) is applied. Then, in the former case, $uu_1 = ABBu'$ and $v_1v = v'BBA$, and, in the latter case, $uu_1 = BABu'$ and $v_1v = v'BAB$, i.e.,

$$SABBu'Sv'BBASwA \Rightarrow Su'Sv'SwA \quad [(\text{IV})] \tag{10}$$

and the derivation proceeds as in (6) or

$$SBABu'Sv'BABSwA \Rightarrow Su'Sv'SwA \quad [(\text{V})] \tag{11}$$

and the derivation also proceeds as in (6) because $u' \in \{ABB, BAB\}^*$ and $v' \in \{BBA, BAB, AaBB : a \in T\}^*$. Notice that the application of a production constructed in (II) would lead, in its consequence, to an incorrect sentential form because the derivation would reach one of the following two forms $SABBxSyAaBBSzA$ or $SBABxSyAaBBSzA$, and each of productions (IV) and (V) would move either A in front of the first S, or at least one B behind the last S. By induction, it implies that the successful derivation proceeds as

$$Su'Sv'SwA \Rightarrow^* SSSwA \Rightarrow w \quad [((\text{III})+(\text{IV})+(\text{V}))^*(\text{VIII})]. \tag{12}$$

Thus, we have proved that the following sequence of productions

$$((\text{IV})+(\text{V}))((\text{II})+(\text{III}))^*((\text{VI})+(\text{VII}))$$

cannot be applied in any successful derivation of G. Therefore, all applications of productions (VI) and (VII) precede any application of productions (IV) and (V), which means that

$$\bar{v} \in \{BBA, BAB\}^*\{AaBB : a \in T\}^*(ABB + \lambda).$$

Finally, by skipping all productions (IV) and (V) in the considered successful derivation $S \Rightarrow^* w$, we have

$$
\begin{aligned}
S \;&\Rightarrow\; SBBASABBSA \quad [\text{(I)}] \\
&\Rightarrow^* SuSvSwA \quad [((\text{II})+(\text{III})+(\text{VI}))^*(\text{VII})(\text{III})^*] \\
&\Rightarrow\; uvw \quad [(\text{VIII})]\,,
\end{aligned}
$$

where $u \in \{ABB, BAB\}^*$, $v \in \{BBA, BAB\}^*$, $u = v^R$ (see **II**), and $w \in T^*$. It is not hard to see that by applications of the corresponding productions constructed in (II) and (III), ignoring productions (VI) and (VII), and applying $S' \to \lambda$ immediately after the last application of productions constructed in (III), we have that $S' \Rightarrow^* w_1 w_2 w$ in G', where $w_1 \in \{A, C\}^*$ and $w_2 \in \{B, D\}^*$ are such that $h(w_1) = u$ and $h(w_2) = v$. As $u = v^R$, we have that $w_1 w_2 w \Rightarrow^* w$ by productions $AB \to \lambda$ and $CD \to \lambda$, which completes the proof. $\qquad\square$

4 Conclusion

This section summarizes the results and open problems concerning the descriptional complexity of scattered context grammars known so far.

One-nonterminal scattered context grammars: It is proved in [8] that scattered context grammars with only one nonterminal (including erasing productions) are not able to generate all context sensitive languages. However, because of the erasing productions, it is an open problem whether they can generate a language which is not context sensitive.

Two-nonterminal scattered context grammars: As far as the authors know, there is no published study concerning the generative power of scattered context grammars with two nonterminals.

Three-nonterminal scattered context grammars: In this paper, we have shown that scattered context grammars with three nonterminals, where no more than nine nonterminals are simultaneously rewritten during any derivation step, characterize the family of recursively enumerable languages. However, no other descriptional complexity measures, such as the number of non-context-free productions, are limited in this paper.

Note that Greibach and Hopcroft [2] have shown that every scattered context grammar can be transformed to an equivalent scattered context grammar where no more than two nonterminals are simultaneously rewritten during any derivation step. This transformation, however, introduces many new nonterminals and, therefore, does not improve our result. Thus, it is an open problem whether the maximal number of nonterminals simultaneously rewritten during any derivation step can be reduced to two in case of scattered context grammars with three nonterminals.

Finally, it is also an open problem whether the number of non-context-free productions can be limited.

Four-nonterminal scattered context grammars: It is proved in [5] that every recursively enumerable language can be generated by a scattered context

grammar with four nonterminals and three non-context-free productions, where no more than six nonterminals are simultaneously rewritten during any derivation step. In comparison with the result of this paper, that result improves the maximal number of simultaneously rewritten symbols and limits the number of non-context-free productions. On the other hand, however, it requires more nonterminals.

Five-nonterminal scattered context grammars: It is proved in [12] that every recursively enumerable language can be generated by a scattered context grammar with five nonterminals and two non-context-free productions, where no more than four nonterminals are simultaneously rewritten during any derivation step. Note that this is the best known bound on the number of non-context-free productions. It is an interesting open problem whether this bound can also be achieved in case of scattered context grammars with three nonterminals.

Scattered context grammars with one non-context-free production: In comparison with the previous result, it is a natural question to ask what is the generative power of scattered context grammars with only one non-context-free production. However, as far as the authors know, this is another very interesting open problem.

Nonerasing scattered context grammars: So far, we have only considered scattered context grammars with erasing productions. However, the most interesting open problem in this investigation area is the question of what is the generative power of nonerasing scattered context grammars. It is not hard to see that they can generate only context sensitive languages. However, it is not known whether nonerasing scattered context grammars are powerful enough to characterize the family of context sensitive languages.

Finally, from the descriptional complexity point of view, it is an interesting challenge for the future research to find out whether some results similar to those proved for scattered context grammars with erasing productions can also be achieved in case of nonerasing scattered context grammars.

Acknowledgements

References

[1] V. Geffert. Context-free-like forms for the phrase-structure grammars. In M. Chytil, L. Janiga, and V. Koubek, editors, *MFCS*, volume 324 of *Lecture Notes in Computer Science*, pages 309–317. Springer, 1988.

[2] S. Greibach and J. Hopcroft. Scattered context grammars. *Journal of Computer and System Sciences*, 3:233–247, 1969.

[3] D. Kolář. Scattered context grammars parsers. In *Proceedings of the 14th International Congress of Cybernetics and Systems of WOCS*, pages 491–500. Wroclaw University of Technology, 2008.

[4] T. Masopust. *Formal Models: Regulation and Reduction.* PhD thesis, Brno University of Technology, Faculty of Information Technology, Brno, 2007. On-line available at the author's web pages.

[5] T. Masopust. On the descriptional complexity of scattered context grammars. *Theoretical Computer Science*, 410(1):108–112, 2009.

[6] A. Meduna. A trivial method of characterizing the family of recursively enumerable languages by scattered context grammars. In *EATCS Bulletin*, pages 104–106. Springer-Verlag, 1995.

[7] A. Meduna. Generative power of three-nonterminal scattered context grammars. *Theoretical Computer Science*, 246:279–284, 2000.

[8] A. Meduna. Terminating left-hand sides of scattered context productions. *Theoretical Computer Science*, 237:423–427, 2000.

[9] A. Meduna and J. Techet. Scattered context grammars that erase nonterminals in a generalized k-limited way. *Acta Informatica*, 45(7):593–608, 2008.

[10] L. Rychnovský. Parsing of context-sensitive languages. In *Proceedings of the 2nd Workshop on Formal Models, WFM 2007*, pages 219–226. Silesian University, Opava, 2007.

[11] A. Salomaa. *Formal languages.* Academic Press, New York, 1973.

[12] Gy. Vaszil. On the descriptional complexity of some rewriting mechanisms regulated by context conditions. *Theoretical Computer Science*, 330:361–373, 2005.