The Computer Journal Advance Access published January 31, 2016

doi:10.1093/comjnl/bxw001

Phrase-Structure Grammars: Normal Forms and Reduction

Zbyněk Křivka[1]*, Alexander Meduna[1] and Petr Zemek[2]

[1]*Faculty of Information Technology, Department of Information Systems, Brno University of Technology, IT4Innovations Centre of Excellence, Božetěchova 2, Brno 612 66, Czech Republic*
[2]*AVG Technologies CZ, s.r.o., Holandská 879/4, Brno 639 00, Czech Republic*
**Corresponding author: krivka@fit.vutbr.cz*

This paper establishes two new normal forms for phrase-structure grammars in which both context-free rules and non-context-free rules are in prescribed forms. In addition, a limit is placed on the number of context-free rules. More specifically, the first form has $2 + n$ context-free rules, where n is the number of terminals. Concerning non-context-free rules, each of them has the form $AB \rightarrow CD$, where A, B, C, D are nonterminals. The second normal form has always only two context-free rules—$S \rightarrow S\#$ and $\# \rightarrow \varepsilon$, where S is the start symbol, # is a nonterminal, and ε is the empty string. Regarding non-context-free rules, each of them is of the form $AB \rightarrow XD$, where A, B, D are nonterminals and X is a nonterminal or a terminal.

Keywords: formal languages; phrase-structure grammars; normal forms

Received 22 March 2015; revised 3 December 2015
Handling editor: Fairouz Kamareddine

1. INTRODUCTION

Formal language theory has always studied how to turn grammars into their equivalent versions with rules satisfying some prescribed forms. These forms, customarily referred to as normal forms, frequently simplify dealing with the grammars in question. More precisely, we can restrict our attention to the grammars in normal forms without affecting their generative power, and a restriction like this usually makes the proofs of results about them easier. To illustrate this advantage, consider phrase-structure grammars in Kuroda normal forms (see [1]). Recall that a grammar is in this form if any rule is in one of these forms:

$$AB \rightarrow CD,\ A \rightarrow BC,\ A \rightarrow a \quad \text{or} \quad A \rightarrow \varepsilon$$

where A, B, C, D are nonterminals, a is a terminal and ε is the empty string (for brevity, we automatically assume that A, B, C, D, a and ε have this meaning throughout the rest of this section). As illustrated by the proofs of Theorems 1.2.5, 1.4.3, 1.4.4 and 1.5.13, Lemmas 2.3.2 and 2.4.3 in [2] and Theorem 4.7.23 in [3], it is often convenient to make use of this form in order to achieve some important results concerning regulated grammars. The present paper continues with this important subject in terms of phrase-structure grammars.

Formal language theory has already achieved several normal forms for phrase-structure grammars (see [1, 4–8]); a survey of some of these forms is to be found in [9, p. 180]. To give a more detailed insight into these forms, we distinguish context-free rules from non-context-free rules in these grammars. In order to clarify what we mean by these two kinds of rules, we consider a grammatical rule as a *context-free rule* if its left-hand side consists of a single nonterminal; otherwise, it is a *non-context-free rule*. Taking into account this distinction, we can next classify all the existing normal forms into the following three groups—(a), (b) and (c).

(a) Non-context-free rules are turned into normal forms while context-free rules are not; the number of any rules is not limited. Indeed, concerning non-context-free rules, there are some transformations that turn any phrase-structure grammar into an equivalent phrase-structure grammar in which all non-context-free rules are in prescribed forms (see Theorem 1.4 on p. 180 in [9], Theorem 9.2 on p. 82 in [10] and Exercise 2 on p. 118 in [10]).
(b) Non-context-free rules are turned into normal forms and their number is restricted; however, the number of context-free rules is not restricted at all. That is, there are some transformations that convert any phrase-structure

grammar to an equivalent phrase-structure grammar in which all non-context-free rules are in prescribed forms, and their number is limited (see [6, 7]). None of these transformations places any restrictions on the number of context-free rules.

(c) There are transformations that turn any phrase-structure grammar into an equivalent phrase-structure grammar in which all rules are in prescribed forms (see [11–13]), such as Kuroda and Penttonen normal forms (see [1, 5]). None of these transformations places any restrictions on the number of any rules.

So far, formal language theory has not achieved any normal forms for phrase-structure grammars so that all rules, including both context-free and non-context-free rules, satisfy some prescribed forms, and in addition, the number of context-free rules is limited. To fill this gap, the present paper establishes the following two normal forms:

(I) First, we explain how to turn any phrase-structure grammar into an equivalent phrase-structure grammar in which every context-free rule is of the form $A \to x$, where x is a terminal, a two-nonterminal string or ε. In addition, the number of context-free rules as well as the number of nonterminals that these rules contain is limited to $2 + n$, where n is the number of terminals. Concerning non-context-free rules, each of them has the form $AB \to CD$.

(II) In the second normal form, phrase-structure grammars have only two context-free rules—that is, the number of context-free rules is reduced independently of the number of terminals as opposed to the first normal form. Specifically, we describe how to turn any phrase-structure grammar into an equivalent phrase-structure grammar that has two context-free rules of the forms $A \to AB$ and $B \to \varepsilon$. Consequently, these rules contain only two nonterminals. Regarding non-context-free rules, they all are of the form $AB \to XD$, where X is a nonterminal or a terminal.

These two new normal forms represent the most important value of the present paper. When investigating phrase-structure grammars, we can always restrict our attention to the grammars that satisfy these forms without affecting their generative power.

The rest of the paper is organized as follows. First, Section 2 gives all the necessary terminology. Then, Section 3 establishes the above-mentioned normal forms for phrase-structure grammars. Finally, Section 4 concludes the paper.

2. PRELIMINARIES

We assume that the reader is familiar with formal language theory (see [9]). For a set Q, card(Q) denotes the cardinality of Q. For an alphabet (finite nonempty set) V, V^* represents the free monoid generated by V under the operation of concatenation. Members of V and V^* are called *symbols* and *strings*, respectively. The identity of V^* is denoted by ε, referred to as the *empty string*. Let $V^+ = V^* - \{\varepsilon\}$; algebraically, V^+ is thus the free semi-group generated by V under the operation of concatenation.

A *phrase-structure grammar* is a quadruple

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

where N and T are two disjoint alphabets, referred to as the alphabet of *nonterminals* and *terminals*, respectively, $S \in N$ is the *start symbol*, and $P \subseteq (N \cup T)^* N (N \cup T)^* \times (N \cup T)^*$ is a finite relation called the set of *rules*. Let $V = N \cup T$ and V is called the *total alphabet*. Each $(x, y) \in P$ is written as $x \to y$ throughout this paper. The *direct derivation relation* over V^*, symbolically denoted by $\Rightarrow$, is defined as follows: $uxv \Rightarrow uyv$ in G if and only if $u, v \in V^*$ and $x \to y \in P$. Let $\Rightarrow^n$ and $\Rightarrow^*$ denote the nth power of $\Rightarrow$, for some $n \geq 0$, and the reflexive-transitive closure of $\Rightarrow$, respectively. The *language of* G is denoted by $L(G)$ and defined as $L(G) = \{w \in T^* \mid S \Rightarrow^* w\}$. Two phrase-structure grammars are *equivalent* if and only if they generate the same language.

Let $G = (N, T, P, S)$ be a phrase-structure grammar. G is in the *Kuroda normal form* (see [1]) if every rule in P is in one of the following four forms:

$$\text{(i) } AB \to CD, \quad \text{(iii) } A \to a$$
$$\text{(ii) } A \to BC, \quad \text{(iv) } A \to \varepsilon,$$

where $A, B, C, D \in N$ and $a \in T$.

LEMMA 2.1 (see [1]). *For every phrase-structure grammar G, there is an equivalent phrase-structure grammar H in the Kuroda normal form.*

3. NEW NORMAL FORMS FOR PHRASE-STRUCTURE GRAMMARS

In this section, we establish two new normal forms for phrase-structure grammars.

THEOREM 3.1. *For any phrase-structure grammar, there is an equivalent phrase-structure grammar having rules $S \to S\#$ and $\# \to \varepsilon$, where S is the start symbol and $\#$ is a newly introduced nonterminal, and each of the other rules is in one of these forms*

$$\text{(i) } AB \to CD \quad \text{or} \quad \text{(ii) } A \to a$$

where A, B, C, D are nonterminals and a is a terminal.

Proof. Let $G = (N, T, P, S)$ be a phrase-structure grammar. By Lemma 2.1, we may assume that G is in the Kuroda normal form. Set $\bar{T} = \{\bar{a} \mid a \in T\}$. Without any loss of generality, we assume

that $N, T, \bar{T}$ and $\{\#\}$ are pairwise disjoint. Construct the phrase-structure grammar

$$H = (N', T, P'_1 \cup P'_2 \cup P'_3, S)$$

as follows. Initially, set $N' = N \cup \bar{T} \cup \{\#\}$, $P'_1 = \{S \to S\#, \# \to \varepsilon\}$ and $P'_2 = \{\bar{a} \to a \mid a \in T\}$. P'_3 is constructed by performing (1) through (5), given next:

(1) for each $AB \to CD \in P$, where $A, B, C, D \in N$, extend P'_3 by adding
$$AB \to CD$$

(2) for each $A \to BC \in P$, where $A, B, C \in N$, extend P'_3 by adding
$$A\# \to BC$$

(3) for each $A \to a \in P$, where $A \in N$ and $a \in T$, extend P'_3 by adding
$$A\# \to \bar{a}\#$$

(4) for each $A \to \varepsilon \in P$, where $A \in N$, extend P'_3 by adding
$$A\# \to \#\#$$

(5) for each $A \in N$, extend P'_3 by adding
$$A\# \to \#A$$

Observe that by performing (1) through (4), for each rule from P, we add one new rule into P'_3. In addition, by performing (5), for every nonterminal from G, we add one new rule into P'_3.

Before proving that $L(H) = L(G)$, let us give an insight into the construction. We simulate G by H using the following sequences of derivation steps.

First, by repeatedly using $S \to S\#$, we generate a proper number of #s. Observe that if the number of #s is too low, the derivation can be blocked since rules from (2) consume # during their application. Furthermore, note that only rules from (4) and the initial rule $S \to S\#$ increase the number of #s in sentential forms of H.

Next, we simulate an application of a rule in G by several derivation steps in H. More specifically, by using rules from (5), we can pass # to the left in the current sentential form at will. Whenever # or B occurs as a neighbor of A, we can apply a rule from (1), (2) and (4). We can also erase any occurrences of # by using $\# \to \varepsilon$.

Then, to simulate rewriting according to rules of the form $A \to a$, we rewrite every occurrence of nonterminal A by nonterminal $\bar{a}$ by the application of rules from (3). Observe that a premature application of a rule of this kind would block the derivation in H because H could not move #s to the left in such a sentential form.

To conclude the simulation, we rewrite the current sentential form by rule $\# \to \varepsilon$ and rules of the form $\bar{a} \to a$ to generate a string of terminals.

To establish the identity $L(H) = L(G)$, we prove four claims. Claim 1 demonstrates that every $w \in L(H)$ can be generated in H in three parts; first, only nonterminals from $N \cup \{\#\}$ are generated, second, nonterminals from N are replaced by nonterminals from $\bar{T}$, and then, all nonterminals are rewritten to terminals by rules from P'_2. Claim 2 shows that we can arbitrarily generate and migrate #s to the left in sentential forms of H during the first part. Claim 3 shows how derivations of G are simulated by H. Finally, Claim 4 shows how derivations of every $w \in L(H)$ in H are simulated by G.

Set $V = N \cup T$ and $V' = N' \cup T$. Define the homomorphism τ from V'^* to V^* as $\tau(X) = X$ for all $X \in V$, $\tau(\bar{a}) = a$ for all $a \in T$, and $\tau(\#) = \varepsilon$.

CLAIM 1. *H can generate every $w \in L(H)$ by this three-part derivation*

$$\begin{aligned} &\text{(i)} \quad S \Rightarrow^* \alpha \\ &\text{(ii)} \quad \Rightarrow^* \beta \\ &\text{(iii)} \quad \Rightarrow^* w, \end{aligned}$$

where a sentential form is in $(N \cup \{\#\})^*$ if and only if it occurs within $S \Rightarrow^* \alpha, \beta \in (\bar{T} \cup \{\#\})^*$ and $w \in T^*$.

Proof. Let $w \in L(H)$. Thus, $S \Rightarrow^* w$. Note that no left-hand side of a rule from $P'_1 \cup P'_3$ contains any symbols from $\bar{T} \cup T$. As a result, we can rearrange the applications of the rules during $S \Rightarrow^* w$ so that

$$S \Rightarrow^* \alpha \Rightarrow^* w$$

whereas only symbols from $(N \cup \{\#\})$ are produced during $S \Rightarrow^* \alpha$ by using rules from $P'_1 \cup P'_3$ excluding rules from (3), so $\alpha \in (N \cup \{\#\})^*$. Take $\alpha \Rightarrow^* w$. We can rearrange this derivation part in the following way. First, we replace all nonterminals from N by using only rules from (3); apart from this, we can erase some #s by $\# \to \varepsilon$. After this, we complete the derivation by repeatedly applying rules from $P'_2 \cup \{\# \to \varepsilon\}$. In this way, we can change $\alpha \Rightarrow^* w$ to

$$\alpha \Rightarrow^* \beta \Rightarrow^* w,$$

where $\beta \in (\bar{T} \cup \{\#\})^*$.

Putting together all the rearranged derivation portions above, we obtain

$$S \Rightarrow^* \alpha \Rightarrow^* \beta \Rightarrow^* w,$$

where $\alpha \in (N \cup \{\#\})^*$, $\beta \in (\bar{T} \cup \{\#\})^*$ and $w \in T^*$. Hence, the claim holds. □

CLAIM 2. *If $S \Rightarrow^* uv$ in H, where $u, v \in V'^*$, then $S \Rightarrow^* u\#v$ in H.*

Proof. First, recall that by the application of $S \to S\#$ there is $S \Rightarrow S\#$ in H. Now, we study two cases.

(A) If $uv \in (N \cup \{\#\})^*$, we continue the derivation from $S\#$ into $uv\#$ and then, by applying rules from (5), # can freely

migrate to the left through the sentential form as needed, so $uv\# \Rightarrow^* u\#v$ in H. Therefore,

$$S \Rightarrow S\# \Rightarrow^* uv\# \Rightarrow^* u\#v$$

in H.

(B) Let $uv \notin (N \cup \{\#\})^*$, so uv contains some symbols from $\bar{T} \cup T$. Let us reconsider Claim 1 and its proof modified so that $w = uv$ (no need to distinguish (ii) and (iii)) and $\alpha = u'v'$ is the last sentential form from $(N \cup \{\#\})^*$ in the derivation of uv in H. Therefore,

$$S \Rightarrow^* \alpha = u'v' \Rightarrow^* uv$$

in H. Observe that by Claim 1 during $u'v' \Rightarrow^* uv$ H applies only rules from (3), P'_2 and $\# \to \varepsilon$. Just like in the conclusion of (A), we apply rules from (5) to obtain

$$S \Rightarrow S\# \Rightarrow^* u'v'\# \Rightarrow^* u'\#v'$$

in H. As the final part of (B), we prove that

$$u'v' \Rightarrow^* uv$$

in H made only by non-context-free rules from (3) and context-free rules from $P'_2 \cup \{\# \to \varepsilon\}$ implies that we can make

$$u'\#v' \Rightarrow^* u\#v$$

in H as well. It is easy to see that rules with their left-hand sides entirely in either u' or v' can be applied in $u'\#v' \Rightarrow^* u\#v$ exactly in the same way as in $u'v' \Rightarrow^* uv$. Note that this holds for all rules from $P'_2 \cup \{\# \to \varepsilon\}$ because these rules are context-free. The last case to study is the application of non-context-free rules from (3) of the form $A\# \to \bar{a}\#$ in $\alpha = u'v'$ with $u' = u''A$ and $v' = \#v''$. As

$$u'v' = u''A\#v'' \Rightarrow u''\bar{a}\#v'' \Rightarrow^* uv$$

in H, then also

$$u'\#v' = u''A\#\#v'' \Rightarrow u''\bar{a}\#\#v'' \Rightarrow^* u\#v$$

in H, so the claim holds. □

CLAIM 3. *If $S \Rightarrow^k x$ in G, where $x \in V^*$, for some $k \geq 0$, then $S \Rightarrow^* x'$ in H, where $\tau(x') = x$.*

Proof. This claim is established by induction on $k \geq 0$.

Basis. Let $k = 0$. That is, $S \Rightarrow^0 x$ in G. By the definition of $\Rightarrow^n$, for $n = 0$, $x = S$. Thus, $S \Rightarrow^0 S$ in G. Note that $S \Rightarrow^0 S$ in H as well.

Induction Hypothesis. For some $k \geq 0$, $S \Rightarrow^k x$ in G implies that $S \Rightarrow^* x'$ in H such that $x = \tau(x')$.

Induction Step. Let $u, v \in N'^*$, $A, B, C, D \in N$ and $m \geq 0$. Assume that $S \Rightarrow^k y \Rightarrow x$ in G. By the induction hypothesis, $S \Rightarrow^* y'$ in H with $y = \tau(y')$. Let us show the simulation of $y \Rightarrow x$ in G by an application of several derivation steps in H to get $y' \Rightarrow^+ x'$ with $\tau(x') = x$. This simulation is divided into the following four cases, (i) through (iv).

(i) Simulation of the application of $AB \to CD$:

$$y' = uA\#^m Bv \Rightarrow^m u\#^m ABv \Rightarrow u\#^m CDv = x'$$

in H using m derivation steps according to rules $A\# \to \#A$ from (5), and concluding the derivation by rule $AB \to CD$ from (1).

Consider the induction hypothesis for $y = \tau(u)A\tau(v)$ to observe that uAv represents a sentential form in H. From this observation, it follows that $y' = uA\#v$ is also a sentential form in H by Claim 2.

(ii) Simulation of the application of $A \to BC$:

$$y' = uA\#v \Rightarrow uBCv = x'$$

in H using rule $A\# \to BC$ from (2);

(iii) simulation of the application of $A \to a$:

$$y' = uA\#v \Rightarrow u\bar{a}\#v = x'$$

in H using rule $A\# \to \bar{a}\#$ from (3);

(iv) simulation of the application of $A \to \varepsilon$:

$$y' = uA\#v \Rightarrow u\#\#v = x'$$

in H using rule $A\# \to \#\#$ from (4).

Thus, the claim holds. □

CLAIM 4. *If $S \Rightarrow^k x'$ in H, where $x' \in V'^*$, for some $k \geq 0$, then $S \Rightarrow^* x$ in G with $x = \tau(x')$.*

Proof. This claim is established by induction on $k \geq 0$.

Basis. For $S \Rightarrow^0 S$ in H, there is $S \Rightarrow^0 S$ in G.

Induction Hypothesis. For some $k \geq 0$, $S \Rightarrow^k x'$ in H implies that $S \Rightarrow^* x$ in G such that $x = \tau(x')$.

Induction Step. Let $u, v \in V'^*$, $A, B, C, D \in N$ and $\bar{a} \in \bar{T}$. Assume that $S \Rightarrow^k y' \Rightarrow x'$ in H. By the induction hypothesis, $S \Rightarrow^* y$ in G such that $y = \tau(y')$. Let us examine the following eight possibilities of $y' \Rightarrow x'$ in H:

Next, we establish statements (i) through (viii). Observe that statements (i), (vi) and (vii) follow from the fact that $\tau(\#) = \varepsilon$.

(i) $y' = uSv \Rightarrow uS\#v = x'$ in H: By zero steps, G performs

$$\begin{array}{c} \tau(y') = y = \tau(uSv) \\ \Rightarrow^0 \\ \tau(uS\#v) = \tau(uSv) = x = \tau(x') \end{array}$$

(ii) $y' = uABv \Rightarrow uCDv = x'$ in H: According to (1), G performs

$$y = \tau(u)AB\tau(v) \Rightarrow \tau(u)CD\tau(v) = x$$

(iii) $y' = uA\#v \Rightarrow uBCv = x'$ in H: According to the source rule in (2), G performs

$$y = \tau(u)A\tau(\#v) \Rightarrow \tau(u)BC\tau(\#v) = \tau(u)BC\tau(v) = x$$

(iv) $y' = uA\#v \Rightarrow u\bar{a}\#v = x'$ in H: According to the source rule $A \to a$ in (3), G performs

$$y = \tau(u)A\tau(\#v) \Rightarrow \tau(u\bar{a}\#v) = \tau(u)a\tau(v) = x$$

(v) $y' = uA\#v \Rightarrow u\#\#v = x'$ in H: By the corresponding rule $A \to \varepsilon$, G performs

$$y = \tau(u)A\tau(\#v) \Rightarrow \tau(u\#\#v) = \tau(uv) = x$$

(vi) $y' = uA\#v \Rightarrow u\#Av = x'$ in H: In G,

$$y = \tau(uA\#v) = \tau(u)A\tau(\#v) \Rightarrow^0 \tau(u\#)A\tau(v) = x$$

(vii) $y' = u\#v \Rightarrow uv = x'$ in H: In G,

$$y = \tau(u\#v) \Rightarrow^0 \tau(uv) = x$$

(viii) $y' = u\bar{a}v \Rightarrow uav = x'$ in H: In G,

$$y = \tau(u\bar{a}v) = \tau(u)a\tau(v) \Rightarrow^0 \tau(u)a\tau(v) = x$$

Thus, the claim holds. □

Next, we establish the identity $L(H) = L(G)$. Consider Claim 3 with $x \in T^*$. Then, $S \Rightarrow^* x$ in G implies that $S \Rightarrow^* x$ in H, so $L(G) \subseteq L(H)$. Consider Claim 4 with $x' \in T^*$. Then, $S \Rightarrow^* x' = \tau(x')$ in H implies that $S \Rightarrow^* x = \tau(x')$ in G, so $L(H) \subseteq L(G)$. Hence, $L(H) = L(G)$.

Observe that H has the form described in Theorem 3.1. Thus, this theorem holds. □

From the construction given in the proof of Theorem 3.1, we obtain the following corollary concerning the number of nonterminals and rules in the resulting grammar.

COROLLARY 3.1. *Let $G = (N, T, P, S)$ be a phrase-structure grammar in the Kuroda normal form. Then, there is an equivalent phrase-structure grammar,*

$$H = (N', T, P', S),$$

which satisfies properties (a), (b) *and* (c), *given next.*

(a) *H satisfies the normal form from Theorem 3.1;*
(b) $\text{card}(N') = \text{card}(N) + \text{card}(T) + 1$;
(c) $\text{card}(P') = \text{card}(P) + \text{card}(T) + \text{card}(N) + 2$.

Proof. Let $G = (N, T, P, S)$ be a phrase-structure grammar in the Kuroda normal form. Introduce a phrase-structure grammar, $H = (N', T, P', S)$, in the following way. Let N', T and S have the same meaning as in the proof of Theorem 3.1. Set $P' = P'_1 \cup P'_2 \cup P'_3$, where P'_1, P'_2 and P'_3 are defined just like in the proof of Theorem 3.1, which implies that H defined in this way is equivalent to G and that H satisfies the normal form from Theorem 3.1. Examine the proof of Theorem 3.1 to see that (b) and (c) hold. □

Consider Corollary 3.1. If we drop the requirement that in the non-context-free rules, each symbol is a nonterminal, then we can reduce the number of context-free rules from $\text{card}(T) + 2$ to two.

THEOREM 3.2. *For any phrase-structure grammar, there is an equivalent phrase-structure grammar having rules $S \to S\#$ and $\# \to \varepsilon$, where S is the start symbol and $\#$ is a newly introduced nonterminal, and each of the other rules is of the form*

$$AB \to XD$$

where A, B, D are nonterminals and X is a nonterminal or terminal.

Proof. Let $G = (N, T, P, S)$ be a phrase-structure grammar. By Lemma 2.1, we may assume that G is in the Kuroda normal form. Reconsider the proof of Theorem 3.1. Observe that we can obtain H in this new normal form by omitting P'_2 and modifying Step (3) in the following way:

(3) for each $A \to a \in P$, where $A \in N$ and $a \in T$, extend P'_3 by adding

$$A\# \to a\#$$

The rest of the proof is analogous to the proof of Theorem 3.1, so it is left to the reader. □

4. CONCLUSION

As already pointed out, the most important value of these results consists in the achievement of new normal forms for phrase-structure grammars so all their rules satisfy some prescribed forms, and in addition, the number of their context-free rules is limited. As a result, in the future, whenever investigating phrase-structure grammars, formal language theory can simplify the investigation by narrowing its attention to the

grammars satisfying these forms without affecting their generative power.

Apart from this key value, the achieved results might be of some other interest, too. They demonstrate that while any reduction of the number of nonterminals in context-free rules is ruled out in terms of some grammars, it is achievable in terms of other grammars. For instance, recall that in terms of context-free grammars, no reduction like this is possible (see [14]). At a glance, one might expect a similar result in terms of the number of nonterminals occurring in context-free rules of phrase-structure grammars. Surprisingly, this is not the case as follows from Theorem 3.2, proved above. Indeed, with only two nonterminals S and # occurring in two context-free rules—that is, $S \to S\#$ and $\# \to \varepsilon$, the phrase-structure grammars keep their generative power unchanged.

By no means, the present paper closes the vivid investigation of normal forms for phrase-structure grammars with a limited number of nonterminals. Perhaps most importantly, so far, formal language theory has not answered whether any phrase-structure grammar can be turned into an equivalent phrase-structure grammar in which all rules are in normal forms and, simultaneously, the number of its nonterminals is limited.

ACKNOWLEDGEMENTS

The authors deeply thank all the three anonymous referees for their invaluable comments and suggestions.

FUNDING

This work has been supported by the IT4IXS—IT4Innovations Excellence in Science project (LQ1602), the BUT FIT FIT-S-14-2299 grant and the TAČR TE01020415 grant.

REFERENCES

[1] Kuroda, S.Y. (1964) Classes of languages and linear-bounded automata. *Inf. Control*, **7**, 207–223.

[2] Dassow, J. and Păun, G. (1989) *Regulated Rewriting in Formal Language Theory*. EATCS Monographs on Theoretical Computer Science 18. Springer, Berlin.

[3] Meduna, A. and Zemek, P. (2014) *Regulated Grammars and Automata*. Springer, New York.

[4] Kolář, D. and Meduna, A. (2002) Homogenous grammars with a reduced number of non-context-free productions. *Inf. Process. Lett.*, **2002**, 253–257.

[5] Penttonen, M. (1974) One-sided and two-sided context in formal grammars. *Inf. Control*, **25**, 371–392.

[6] Geffert, V. (1988) Context-Free-Like Forms for the Phrase-structure Grammars. *Proc. Mathematical Foundations of Computer Science'88*, Carlsbad, Czechoslovakia, August 29–September 2, pp. 309–317. Springer, Berlin.

[7] Geffert, V. (1991) Normal forms for phrase-structure grammars. *RAIRO Inf. Théor. Appl.*, **25**, 473–496.

[8] Révész, G. (2012) *Introduction to Formal Languages*. Dover Publications, New York.

[9] Rozenberg, G. and Salomaa, A. (eds) (1997) *Handbook of Formal Languages, Vol. 1: Word, Language, Grammar*. Springer, New York.

[10] Salomaa, A. (1973) *Formal Languages*. Academic Press, London.

[11] Smith, W.B. (1970) Error detection in formal languages. *J. Comput. Syst. Sci.*, **4**, 385–405.

[12] Révész, G. (1974) Comment on the paper "Error detection in formal languages". *J. Comput. Syst. Sci.*, **8**, 238–242.

[13] Penttonen, M. (1972) A normal form for context-sensitive grammars. *Ann. Univ. Turku. Ser. AI*, **156**, 1–12.

[14] Gruska, J. (1969) Some classifications of context-free languages. *Inf. Control*, **14**, 152–179.