Theoretical Computer Science 410 (2009) 1961–1969

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

An infinite hierarchy of language families generated by scattered context grammars with n-limited derivations

Alexander Meduna*, Jiří Techet

Department of Information Systems, Faculty of Information Technology, Brno University of Technology, Božetěchova 2, Brno 61266, Czech Republic

ARTICLE INFO

Article history:
Received 24 January 2008
Received in revised form 27 August 2008
Accepted 8 December 2008
Communicated by M. Ito

Keywords:
Scattered context grammars
Unordered scattered context grammars
Left derivation restriction
Generative power
Infinite hierarchy of language families

ABSTRACT

This paper introduces scattered context grammars without erasing productions, in which an application of a production always occurs within the first n nonterminals of the current sentential form. It demonstrates that this restriction gives rise to an infinite hierarchy of language families each of which is properly included in the family of context-sensitive languages. In addition, it proves analogous results for unordered scattered context grammars. Some consequences of these results are derived and open problems formulated.
© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Scattered context grammars were introduced in [2]. These grammars are based upon finite sets of sequences of context-free productions, each of which has a single nonterminal and a non-empty word on its left-hand side and the right-hand side, respectively (scattered context grammars generalized by allowing the empty word to be the right-hand side of a production are not considered in this paper at all). During every derivation step, these grammars simultaneously rewrite k nonterminals in the current sentential form according to a sequence of k context-free productions in the order corresponding to the appearance of these productions in the sequence. It is well known that the language family that these grammars generate is a proper superfamily of the context-free language family. On the other hand, this family is contained in the context-sensitive language family, but it is a longstanding open problem whether this containment is proper.

There exist several modified versions of scattered context grammars (see [1,5,9]), unordered scattered context grammars are one of them (see [6,10]). Unlike ordinary scattered context grammars, these unordered versions may apply a production to any permutation of nonterminals appearing on the left-hand side of the production. The family of languages generated by these grammars is identical to the family of languages generated by matrix grammars without appearance checking, which is properly included in the family of languages generated by scattered context grammars without erasing productions (see [3,10]).

As the formal language theory has always introduced and studied various left restrictions placed on grammatical derivations, we investigate this classical topic in terms of scattered context grammars in the present paper. More specifically, we discuss the language families generated by scattered context grammars that use n-limited derivations, where n is a positive integer. In these derivations, a scattered context production is always applied within the first n occurrences of nonterminals in the current sentential form. We demonstrate that this restriction gives rise to an infinite hierarchy of

* Corresponding author. Tel.: +420 54114 1232.
E-mail address: meduna@fit.vutbr.cz (A. Meduna).

0304-3975/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2008.12.013

language families, each of which is properly included in the family of context-sensitive languages. In addition, we prove analogous results in terms of unordered scattered context grammars as well.

Based upon the proper inclusion, we formulate several consequences open problems. Perhaps most importantly, we point out that the language family generated by the scattered context grammars that make derivation in the n-limited way is properly contained in the context-sensitive language family, so in this sense, we partially contribute to the solving of the longstanding open problem pointed out above.

2. Preliminaries and definitions

We assume that the reader is familiar with formal language theory (see [7,11]). For an alphabet V, $|V|$ denotes the cardinality of V. V^* represents the free monoid generated by V. The unit of V^* is denoted by ε. Set $V^+ = V^* - \{\varepsilon\}$. For $w \in V^*$, $|w|$ and $\text{alph}(w)$ denote the length of w and the set of symbols occurring in w, respectively. For $W \subseteq V$, $|w|_W$ denotes the number of occurrences of elements from W in w. The set of all permutations of $\{1, \dots, n\}$ is denoted by $\text{perm}(n)$. For all $x_1, \dots, x_n \in V^*$ and $(i_1, \dots, i_n) \in \text{perm}(n)$, set $\text{reorder}((x_1, \dots, x_n), (i_1, \dots, i_n)) = (x_{i_1}, \dots, x_{i_n})$.

A *state grammar* (see [4]) is a sixtuple $G = (V, T, K, P, S, p_0)$, where V is an alphabet, $T \subset V$, K is a finite set of states, $S \in V - T$, $p_0 \in K$, and P is a finite set of productions of the form $(A, p) \to (x, q)$, where $A \in V - T$, $x \in V^+$, and $p, q \in K$. If $u = (rAs, p)$, $v = (rxs, q)$, and $(A, p) \to (x, q) \in P$, where $r, s \in V^*$, and for each $(B, p) \to (y, t) \in P$, where $B \in V - T$, $y \in V^+$, $t \in K$, it holds that $B \notin \text{alph}(r)$, then G makes a derivation step from u to v according to $(A, p) \to (x, q)$, symbolically written as $u \Rightarrow_G v\ [(A, p) \to (x, q)]$ or, simply, $u \Rightarrow_G v$. To emphasize that the jth nonterminal in u is rewritten during a derivation step, we write $u\ {}^j{\Rightarrow_G}\ v$. The *state language* is a language generated by a state grammar G, denoted by $L(G)$, and defined as $L(G) = \{x \in T^* \ :\ (S, p_0) \Rightarrow_G^* (x, q) \text{ for some } q \in K\}$. The family of all state languages is denoted by $\mathscr{L}(ST)$. An *n-limited derivation*, denoted by $x\ {}^n{\Rightarrow}_G^*\ y$, is a derivation in which every derivation step $u\ {}^j{\Rightarrow_G}\ v$ satisfies $j \le n$. Set $L(G, n) = \{x \in T^* \ :\ (S, p_0)\ {}^n{\Rightarrow}_G^*\ (x, q) \text{ for some } q \in K\}$. A state grammar G is *of order n* if and only if $L(G, n) = L(G)$. A state language L is said to be of order n if and only if $L = L(G, n)$, for a state grammar G. The family of state languages of order n is denoted by $\mathscr{L}(ST, n)$. Set $\mathscr{L}(ST, \infty) = \bigcup_{i=1}^{\infty} \mathscr{L}(ST, i)$.

A *scattered context grammar* (see [2]) is defined as a quadruple $G = (V, T, P, S)$, where V is an alphabet, $T \subset V$, $S \in V - T$, and P is a finite set of productions such that every production has the form $(A_1, \dots, A_k) \to (x_1, \dots, x_k)$ for some $k \ge 1$, where $A_i \in V - T$, $x_i \in V^*$ for all $1 \le i \le k$. If every production $(A_1, \dots, A_k) \to (x_1, \dots, x_k) \in P$ satisfies $x_i \in V^+$ for all $1 \le i \le k$, G is a *propagating scattered context grammar*. If $p = (A_1, \dots, A_k) \to (x_1, \dots, x_k) \in P$, $u = u_1A_1u_2 \dots u_kA_ku_{k+1}$, and $v = u_1x_1u_2 \dots u_kx_ku_{k+1}$, where $u_i \in V^*$ for all $1 \le i \le k+1$, then G makes a *derivation step* from u to v according to p, symbolically written as $u \Rightarrow_G v\ [p]$ or, simply, $u \Rightarrow_G v$. The language of G is denoted by $L(G)$ and defined as $L(G) = \{x \in T^* \ :\ S \Rightarrow_G^* x\}$. The family of languages generated by propagating scattered context grammars is denoted by $\mathscr{L}(PSC)$. If $u = u_1A_1 \dots u_kA_ku_{k+1} \Rightarrow_G u_1x_1 \dots u_kx_ku_{k+1} = v$ and $|u_1A_1 \dots u_kA_k|_{V-T} \le n$, then the derivation step is *n-limited* and we write $u\ {}^n{\Rightarrow_G}\ v$. An *$n$-limited derivation*, denoted by $x\ {}^n{\Rightarrow}_G^*\ y$, is a derivation in which every derivation step $u\ {}^j{\Rightarrow_G}\ v$ satisfies $j \le n$. Define the *language of order n* generated by G as $L(G, n) = \{x \in T^* \ :\ S\ {}^n{\Rightarrow}_G^*\ x\}$. The family of languages of order n generated by propagating scattered context grammars is denoted by $\mathscr{L}(PSC, n)$, and $\mathscr{L}(PSC, \infty) = \bigcup_{i=1}^{\infty} \mathscr{L}(PSC, i)$.

An *unordered scattered context grammar* is a quadruple $G = (V, T, P, S)$, where V, T, P, and S are defined as in the case of a scattered context grammar. Analogously, we define a propagating unordered scattered context grammar. If there exists a permutation $\pi \in \text{perm}(k)$, for some $k \ge 1$, such that $p = \text{reorder}((A_1, \dots, A_k), \pi) \to \text{reorder}((x_1, \dots, x_k), \pi) \in P$, and $u = u_1A_1u_2A_2 \dots u_kA_ku_{k+1}$, $v = u_1x_1u_2x_2 \dots u_kx_ku_{k+1}$, where $u_i \in V^*$ for all $1 \le i \le k+1$, then G makes a derivation step from u to v according to p. The generated language, n-limited derivation and language of order n are defined like for scattered context grammars. The family of languages of order n generated by propagating unordered scattered context grammars is denoted by $\mathscr{L}(PUSC, n)$, and $\mathscr{L}(PUSC, \infty) = \bigcup_{i=1}^{\infty} \mathscr{L}(PUSC, i)$.

3. Main result

In this section, we prove $\mathscr{L}(PSC, n) = \mathscr{L}(ST, n)$, for all $n \ge 1$. First, Lemma 1 demonstrates $\mathscr{L}(ST, n) \subseteq \mathscr{L}(PSC, n)$. Then, Lemma 2 shows that $\mathscr{L}(PSC, n) \subseteq \mathscr{L}(ST, n)$.

Lemma 1. *$\mathscr{L}(ST, n) \subseteq \mathscr{L}(PSC, n)$ for all $n \ge 1$.*

Proof. Let $G = (V, T, K, P, S, p_0)$ be a state grammar of order n. Set

$$N_1 = \{\langle A, p, k\rangle \ :\ A \in V - T, p \in K, 1 \le k \le n\},$$
$$N_2 = \{\langle \hat{A}, p, k\rangle \ :\ A \in V - T, p \in K, 1 \le k \le n\},$$
$$N_3 = \{\langle A', p, n-1\rangle \ :\ A \in V - T, p \in K\},$$
$$N_4 = \{\hat{A} \ :\ A \in V - T\}.$$

Set $\alpha(p) = \{A \ :\ (A, p) \to (x, q) \in P\}$ for every $p \in K$. Define the propagating scattered context grammar

$$\bar{G} = (V \cup N_1 \cup N_2 \cup N_3 \cup N_4 \cup \{\bar{S}\}, T, \bar{P}, \bar{S})$$

with $\bar{P}$ constructed as follows (throughout the construction, we add intuitive explanation of the purpose of the constructed productions):

(1) Add $(\bar{S}) \to (\langle \hat{S}, p_0, 1\rangle)$ to $\bar{P}$;
(2) For every $A_1, \ldots, A_k \in V - T$, where $1 \le k \le n$, every

$$(A_r, p) \to (x_1 B_1 \ldots x_t B_t x_{t+1}, q) \in P,$$

where $1 \le r \le k$, $B_1, \ldots, B_t \in V - T$, $x_1, \ldots, x_{t+1} \in T^*$ for some $t \ge 0$, $A_i \notin \alpha(p)$ for every $1 \le i < r$ ($A_1, \ldots, A_k$ denote the first k nonterminals in the sentential form; k is the number of nonterminals present in the sentential form if it contains less than n nonterminals, otherwise $k = n$; A_r is the nonterminal of the sentential form whose rewriting is simulated; t is the number of nonterminals appearing on the right-hand side of the simulated production),

(a) and $r + t - 1 > n$, add

(i) (used when the sentential form contains more than n nonterminals)

$$\begin{aligned}&(\langle A_1, p, n\rangle, \ldots, \langle A_{r-1}, p, n\rangle, \langle A_r, p, n\rangle, \langle A_{r+1}, p, n\rangle, \ldots, \langle A_n, p, n\rangle)\\ \to\ &(\langle A_1, q, n\rangle, \ldots, \langle A_{r-1}, q, n\rangle, x_1\langle B_1, q, n\rangle \ldots x_{n-r+1}\langle B_{n-r+1}, q, n\rangle\\ &\ x_{n-r+2}B_{n-r+2} \ldots x_t B_t x_{t+1}, A_{r+1}, \ldots, A_n)\end{aligned}$$

to $\bar{P}$;

(ii) (used when the sentential form contains at most n nonterminals and A_r is not the last nonterminal) if $r < k$, add

$$\begin{aligned}&(\langle A_1, p, k\rangle, \ldots, \langle A_{r-1}, p, k\rangle, \langle A_r, p, k\rangle, \langle A_{r+1}, p, k\rangle, \ldots, \langle A_{k-1}, p, k\rangle, \langle \hat{A}_k, p, k\rangle)\\ \to\ &(\langle A_1, q, n\rangle, \ldots, \langle A_{r-1}, q, n\rangle, x_1\langle B_1, q, n\rangle \ldots x_{n-r+1}\langle B_{n-r+1}, q, n\rangle\\ &\ x_{n-r+2}B_{n-r+2} \ldots x_t B_t x_{t+1}, A_{r+1}, \ldots, A_{k-1}, \hat{A}_k)\end{aligned}$$

to $\bar{P}$;

(iii) (used when the sentential form contains at most n nonterminals and A_r is the last nonterminal) if $r = k$, add

$$\begin{aligned}&(\langle A_1, p, k\rangle, \ldots, \langle A_{k-1}, p, k\rangle, \langle \hat{A}_k, p, k\rangle)\\ \to\ &(\langle A_1, q, n\rangle, \ldots, \langle A_{k-1}, q, n\rangle, x_1\langle B_1, q, n\rangle \ldots x_{n-k+1}\langle B_{n-k+1}, q, n\rangle\\ &\ x_{n-k+2}B_{n-k+2} \ldots x_{t-1}B_{t-1}x_t \hat{B}_t x_{t+1})\end{aligned}$$

to $\bar{P}$;

(b) and $r + t - 1 \le n$, $k + t - 1 > n$, add

(i) (used when the sentential form contains more than n nonterminals)

$$\begin{aligned}&(\langle A_1, p, n\rangle, \ldots, \langle A_{r-1}, p, n\rangle, \langle A_r, p, n\rangle,\\ &\ \langle A_{r+1}, p, n\rangle, \ldots, \langle A_{n-t+1}, p, n\rangle, \langle A_{n-t+2}, p, n\rangle, \ldots, \langle A_n, p, n\rangle)\\ \to\ &(\langle A_1, q, n\rangle, \ldots, \langle A_{r-1}, q, n\rangle, x_1\langle B_1, q, n\rangle \ldots x_t\langle B_t, q, n\rangle x_{t+1},\\ &\ \langle A_{r+1}, q, n\rangle, \ldots, \langle A_{n-t+1}, q, n\rangle, A_{n-t+2}, \ldots, A_n)\end{aligned}$$

to $\bar{P}$;

(ii) (used when the sentential form contains at most n nonterminals and A_r is not the last nonterminal) if $r < k$, add

$$\begin{aligned}&(\langle A_1, p, k\rangle, \ldots, \langle A_{r-1}, p, k\rangle, \langle A_r, p, k\rangle,\\ &\ \langle A_{r+1}, p, k\rangle, \ldots, \langle A_{n-t+1}, p, k\rangle, \langle A_{n-t+2}, p, k\rangle, \ldots, \langle A_{k-1}, p, k\rangle, \langle \hat{A}_k, p, k\rangle)\\ \to\ &(\langle A_1, q, n\rangle, \ldots, \langle A_{r-1}, q, n\rangle, x_1\langle B_1, q, n\rangle \ldots x_t\langle B_t, q, n\rangle x_{t+1},\\ &\ \langle A_{r+1}, q, n\rangle, \ldots, \langle A_{n-t+1}, q, n\rangle, A_{n-t+2}, \ldots, A_{k-1}, \hat{A}_k)\end{aligned}$$

to $\bar{P}$;

(c) and $k + t - 1 \le n$, and

(i) if $t = 0$, add

(A) (used when the sentential form contains more than n nonterminals and A_r is rewritten to $x_1 \in T^*$)

$$\begin{aligned}&(\langle A_1, p, n\rangle, \ldots, \langle A_{r-1}, p, n\rangle, \langle A_r, p, n\rangle, \langle A_{r+1}, p, n\rangle, \ldots, \langle A_n, p, n\rangle)\\ \to\ &(\langle A'_1, q, n-1\rangle, \ldots, \langle A'_{r-1}, q, n-1\rangle, x_1, \langle A'_{r+1}, q, n-1\rangle, \ldots, \langle A'_n, q, n-1\rangle),\end{aligned}$$

(B) (used immediately after (2.c.i.A))

$$\begin{aligned}&(\langle A'_1, q, n-1\rangle, \ldots, \langle A'_{r-1}, q, n-1\rangle, \langle A'_{r+1}, q, n-1\rangle, \ldots, \langle A'_n, q, n-1\rangle, A_{n+1})\\ \to\ &(\langle A_1, q, n\rangle, \ldots, \langle A_{r-1}, q, n\rangle, \langle A_{r+1}, q, n\rangle, \ldots, \langle A_n, q, n\rangle, \langle A_{n+1}, q, n\rangle),\end{aligned}$$

where $A_{n+1} \in (V - T) \cup N_4$, to $\bar{P}$;

(ii) (used when the sentential form contains at most n nonterminals and A_r is not the last nonterminal) if $r < k$, add

$$\begin{aligned}&(\langle A_1, p, k\rangle, \ldots, \langle A_{r-1}, p, k\rangle, \langle A_r, p, k\rangle, \langle A_{r+1}, p, k\rangle, \ldots, \langle A_{k-1}, p, k\rangle, \langle \hat{A}_k, p, k\rangle)\\ \to\ &(\langle A_1, q, k+t-1\rangle, \ldots, \langle A_{r-1}, q, k+t-1\rangle, x_1\langle B_1, q, k+t-1\rangle \ldots x_t\langle B_t, q, k+t-1\rangle x_{t+1},\\ &\ \langle A_{r+1}, q, k+t-1\rangle, \ldots, \langle A_{k-1}, q, k+t-1\rangle, \langle \hat{A}_k, q, k+t-1\rangle)\end{aligned}$$

to $\bar{P}$;

(iii) (used when the sentential form contains at most n nonterminals and A_r is the last nonterminal)
if $r = k$
(A) and $k > 1$ or $t \neq 0$, add

$$\begin{aligned}&(\langle A_1, p, k\rangle, \ldots, \langle A_{k-1}, p, k\rangle, \langle \hat{A}_k, p, k\rangle)\\ \rightarrow\ &(\langle A_1, q, k+t-1\rangle, \ldots, \langle A_{k-1}, q, k+t-1\rangle,\\ &x_1\langle B_1, q, k+t-1\rangle \ldots x_{t-1}\langle B_{t-1}, q, k+t-1\rangle x_t\langle \hat{B}_t, q, k+t-1\rangle x_{t+1})\end{aligned}$$

to $\bar{P}$;
(B) (simulates the last derivation step of G)
and $k = 1$, $t = 0$, add
$(\langle \hat{A}_1, p, 1\rangle) \rightarrow (x_1)$ to $\bar{P}$.

Basic Idea: For every sentential form of G, (x, p), words u and v can be selected so that $x = uv$ and either $|u|_{V-T} = n$ and $|v|_{V-T} \geq 1$ or $|u|_{V-T} = k$, $k \leq n$ and $|v|_{V-T} = 0$. As a result, only the nonterminals occurring in u can be rewritten by a production of G. As n is a finite number, it is possible to construct a scattered context grammar $\bar{G}$ that rewrites all nonterminals present in u in every derivation step. In this way, we can simulate the rewriting of the leftmost nonterminal for a given state by considering all possible forms of u and constructing productions of $\bar{G}$ accordingly. The constructed grammar simulates every sentential form of G by dividing it into two parts. The first part contains only nonterminals from $N_1 \cup N_2$, which can be rewritten by the constructed productions. The other part contains nonterminals from $(V - T) \cup N_4$, which no production rewrites (with the exception of (2.c.i.B)).

By rewriting a nonterminal in the first part, the number of nonterminals appearing in the first part might exceed n. To prevent this situation, the constructed productions move the extra nonterminals from the end of the first part to the beginning of the second part (see all productions from (2.a) and (2.b)) so the number of nonterminals in the first part is no more than n.

Apart from adding nonterminals, a nonterminal can be removed from the first part as well. This happens when a nonterminal is rewritten to a string over T. In this case, a special action is in order when the second part contains some nonterminals. That is, for this purpose, the grammar records the last nonterminal of the sentential form. If the last nonterminal appears within the first part, it is represented by a symbol from N_2 and if it is located in the second part, it is represented by a symbol from N_4. If a nonterminal from the first part is rewritten to a string over T and the second part contains some nonterminals (that is, a symbol from N_2 does not appear at the end of the first part), the first nonterminal of the second part is removed, converted to a symbol from N_1 (or N_2 if it is the last nonterminal) and added at the end of the first part (see productions (2.c.i.A) and (2.c.i.B)). In this way, the number of nonterminals that appear in the first part remains n. If a symbol from N_2 appears at the end of the first part, the second part can be ignored because it does not contain any nonterminal. In this way the grammar guarantees that if the first part of the sentential form contains less than n nonterminals, the second part does not contain any nonterminal at all. Productions from (2.c.ii) and (2.c.iii) are used when the second part of the sentential form remains empty after the rewriting.

Every production changes the current state p contained within every nonterminal of the first part to the new state q. In addition, each of these nonterminals records the number of nonterminals, k, present in the first part of the sentential form and this number is updated after every derivation step. Therefore, productions that simulate the rewriting in a different state and productions that rewrite a different number of nonterminals are not applicable.

Formal Proof: By examining the constructed productions, we see that the derivations of G and $\bar{G}$ are very similar. Indeed, in most cases, one production of $\bar{G}$ simulates one production of G. However, when a production from (2.c.i.A) is applied, it has to be followed by (2.c.i.B), so in this case, one derivation step in G corresponds to two derivation steps in $\bar{G}$. Formally, we define the term *sf-correspondence* between the sentential *f* orms of G and $\bar{G}$ by the following recursive definition and use this term in the formulation of Claim 1:

(1) The sentential form (S, p_0) of G sf-corresponds to the sentential form $\langle \hat{S}, p_0, 1\rangle$ in $\bar{G}$;
(2) Let $(x, p) \Rightarrow_G (y, q)\ [\alpha]$, where (x, p) sf-corresponds to some $\bar{x}$ in $\bar{G}$.
- If $(x, p) \Rightarrow_G (y, q)\ [\alpha]$ satisfies $|x|_{V-T} > n$, $k + t - 1 \leq n$, and $t = 0$, then (y, q) sf-corresponds to $\bar{z}$ in $\bar{G}$, where $\bar{x} \Rightarrow^2_{\bar{G}} \bar{z}\ [\bar{\alpha}_1\bar{\alpha}_2]$, $\bar{\alpha}_1$ and $\bar{\alpha}_2$ are productions from (2.c.i.A) and (2.c.i.B), respectively, whose construction is based on α;
- otherwise, (y, q) sf-corresponds to $\bar{y}$ in $\bar{G}$, where $\bar{x} \Rightarrow_{\bar{G}} \bar{y}\ [\bar{\alpha}]$ and the construction of $\bar{\alpha}$ is based on α.

Claim 1. *Every sentential form of G, $(y_1A_1 \ldots y_mA_my_{m+1}, p)$, where $p \in K$, $y_1, \ldots, y_{m+1} \in T^*$, $A_1, \ldots, A_m \in V - T$, for some $m \geq 0$, sf-corresponds to one of the following sentential forms in $\bar{G}$:*

(1) *For $m \leq n$, $y_1\langle A_1, p, m\rangle \ldots y_{m-1}\langle A_{m-1}, p, m\rangle y_m\langle \hat{A}_m, p, m\rangle y_{m+1}$;*
(2) *For $m > n$, $y_1\langle A_1, p, n\rangle \ldots y_n\langle A_n, p, n\rangle y_{n+1}A_{n+1} \ldots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}$.*

Proof. Every derivation in $\bar{G}$ starts by the production from step (1) of the construction and this production is not used during the rest of the derivation process, so

$$S \Rightarrow_{\bar{G}} \langle \hat{S}, p_0, 1\rangle.$$

The rest of the claim is proved by induction on length h of derivations, for $h \geq 0$.

Basis. Let $h = 0$. Then, $(S, p_0) \Rightarrow^0_G (S, p_0)$ corresponds to $\langle \hat{S}, p_0, 1\rangle \Rightarrow^0_{\bar{G}} \langle \hat{S}, p_0, 1\rangle$.

Induction Hypothesis. Suppose that the claim holds for all derivations of length h or less, for some $h \geq 0$.

Induction Step. First, consider a sentential form of G, $(y_1A_1 \dots y_mA_my_{m+1}, p)$, where $m \leq n$, and a production

$$(A_r, p) \to (x_1B_1 \dots x_tB_tx_{t+1}, q) \in P,$$

where $1 \leq r \leq m$, $B_1, \dots, B_t \in V - T$, $x_1, \dots, x_{t+1} \in T^*$, for some $t \geq 0$, which is applicable to the above sentential form (that is, $A_i \notin \alpha(p)$ for every $1 \leq i < r$). Then,

$$\begin{aligned} &(y_1A_1 \dots y_{r-1}A_{r-1}y_rA_ry_{r+1}A_{r+1} \dots y_mA_my_{m+1}, p)\\ \Rightarrow_G\ &(y_1A_1 \dots y_{r-1}A_{r-1}y_rx_1B_1 \dots x_tB_tx_{t+1}y_{r+1}A_{r+1} \dots y_mA_my_{m+1}, q). \end{aligned}$$

By the induction hypothesis, for $m \leq n$, the sentential form of $\bar{G}$ sf-corresponding to

$$(y_1A_1 \dots y_{r-1}A_{r-1}y_rA_ry_{r+1}A_{r+1} \dots y_mA_my_{m+1}, p)$$

is of the form

$$y_1\langle A_1, p, m\rangle \dots y_r\langle A_r, p, m\rangle \dots y_{m-1}\langle A_{m-1}, p, m\rangle y_m\langle \hat{A}_m, p, m\rangle y_{m+1}.$$

Now, one of the productions from steps (2.a.ii), (2.a.iii), (2.b.ii), (2.c.ii), and (2.c.iii) is applicable, depending on the simulated production, m, and n:

(1) If $r + t - 1 > n$ and $r < m$, then a production introduced by (2.a.ii) is applied, so

$$\begin{aligned} &y_1\langle A_1, p, m\rangle \dots y_{r-1}\langle A_{r-1}, p, m\rangle y_r\langle A_r, p, m\rangle\\ &y_{r+1}\langle A_{r+1}, p, m\rangle \dots y_{m-1}\langle A_{m-1}, p, m\rangle y_m\langle \hat{A}_m, p, m\rangle y_{m+1}\\ {}_{\lim}^{\ n}\!\!\Rightarrow_{\bar{G}}\ &y_1\langle A_1, q, n\rangle \dots y_{r-1}\langle A_{r-1}, q, n\rangle y_rx_1\langle B_1, q, n\rangle \dots x_{n-r+1}\langle B_{n-r+1}, q, n\rangle x_{n-r+2}B_{n-r+2} \dots x_tB_tx_{t+1}\\ &y_{r+1}A_{r+1} \dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}. \end{aligned}$$

(2) If $r + t - 1 > n$ and $r = m$, then a production introduced by (2.a.iii) is applied, so

$$\begin{aligned} &y_1\langle A_1, p, m\rangle \dots y_{m-1}\langle A_{m-1}, p, m\rangle y_m\langle \hat{A}_m, p, m\rangle y_{m+1}\\ {}_{\lim}^{\ n}\!\!\Rightarrow_{\bar{G}}\ &y_1\langle A_1, q, n\rangle \dots y_{m-1}\langle A_{m-1}, q, n\rangle y_mx_1\langle B_1, q, n\rangle \dots x_{n-m+1}\langle B_{n-m+1}, q, n\rangle\\ &x_{n-m+2}B_{n-m+2} \dots x_{t-1}B_{t-1}x_t\hat{B}_tx_{t+1}. \end{aligned}$$

(3) If $r + t - 1 \leq n$, $m + t - 1 > n$, and $r < m$, then a production introduced by (2.b.ii) is applied, so

$$\begin{aligned} &y_1\langle A_1, p, m\rangle \dots y_{r-1}\langle A_{r-1}, p, m\rangle y_r\langle A_r, p, m\rangle\\ &y_{r+1}\langle A_{r+1}, p, m\rangle \dots y_{m-1}\langle A_{m-1}, p, m\rangle y_m\langle \hat{A}_m, p, m\rangle y_{m+1}\\ {}_{\lim}^{\ n}\!\!\Rightarrow_{\bar{G}}\ &y_1\langle A_1, q, n\rangle \dots y_{r-1}\langle A_{r-1}, q, n\rangle y_rx_1\langle B_1, q, n\rangle \dots x_t\langle B_t, q, n\rangle x_{t+1}\\ &y_{r+1}\langle A_{r+1}, q, n\rangle \dots y_{n-t+1}\langle A_{n-t+1}, q, n\rangle y_{n-t+2}A_{n-t+2} \dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}. \end{aligned}$$

(4) If $m + t - 1 \leq n$ and $r < m$, then a production introduced by (2.c.ii) is applied, so

$$\begin{aligned} &y_1\langle A_1, p, m\rangle \dots y_{r-1}\langle A_{r-1}, p, m\rangle y_r\langle A_r, p, m\rangle\\ &y_{r+1}\langle A_{r+1}, p, m\rangle \dots y_{m-1}\langle A_{m-1}, p, m\rangle y_m\langle \hat{A}_m, p, m\rangle y_{m+1}\\ {}_{\lim}^{\ n}\!\!\Rightarrow_{\bar{G}}\ &y_1\langle A_1, q, m+t-1\rangle \dots y_{r-1}\langle A_{r-1}, q, m+t-1\rangle y_r\\ &x_1\langle B_1, q, m+t-1\rangle \dots x_t\langle B_t, q, m+t-1\rangle x_{t+1}\\ &y_{r+1}\langle A_{r+1}, q, m+t-1\rangle \dots y_{m-1}\langle A_{m-1}, q, m+t-1\rangle y_m\langle \hat{A}_m, q, m+t-1\rangle y_{m+1}. \end{aligned}$$

(5) If $m + t - 1 \leq n$, $r = m$, and $m > 1$ or $t \neq 0$, then a production introduced by (2.c.iii.A) is applied, so

$$\begin{aligned} &y_1\langle A_1, p, m\rangle \dots y_{m-1}\langle A_{m-1}, p, m\rangle y_m\langle \hat{A}_m, p, m\rangle y_{m+1}\\ {}_{\lim}^{\ n}\!\!\Rightarrow_{\bar{G}}\ &y_1\langle A_1, q, m+t-1\rangle \dots y_{m-1}\langle A_{m-1}, q, m+t-1\rangle y_m\\ &x_1\langle B_1, q, m+t-1\rangle \dots x_{t-1}\langle B_{t-1}, q, m+t-1\rangle x_t\langle \hat{B}_t, q, m+t-1\rangle x_{t+1}y_{m+1}. \end{aligned}$$

(6) If $m = 1$, $t = 0$, then a production introduced by (2.c.iii.B) is applied, so

$$y_1\langle \hat{A}_1, p, 1\rangle y_2 \ {}_{\lim}^{\ n}\!\!\Rightarrow_{\bar{G}} y_1x_1y_2.$$

Because this production removes the last symbol from $N_1 \cup N_2$ from the sentential form and this symbol appears on the left-hand side of every production introduced in step (2), this production can be used only during the very last derivation step.

Second, consider a sentential form of G, $(y_1A_1\dots y_mA_my_{m+1}, p)$, where $m > n$, and a production

$$(A_r, p) \to (x_1B_1\dots x_tB_tx_{t+1}, q) \in P,$$

where $1 \le r \le m$, $B_1, \dots, B_t \in V - T$, $x_1, \dots, x_{t+1} \in T^*$ for some $t \ge 0$, which is applicable to the above sentential form. Then,

$$\begin{aligned} &(y_1A_1\dots y_{r-1}A_{r-1}y_rA_ry_{r+1}A_{r+1}\dots y_nA_n\dots y_mA_my_{m+1}, p)\\ \Rightarrow_G\ &(y_1A_1\dots y_{r-1}A_{r-1}y_rx_1B_1\dots x_tB_tx_{t+1}y_{r+1}A_{r+1}\dots y_nA_n\dots y_mA_my_{m+1}, q).\end{aligned}$$

By the induction hypothesis, for $m > n$, the sentential form of $\bar{G}$ sf-corresponding to

$$(y_1A_1\dots y_{r-1}A_{r-1}y_rA_ry_{r+1}A_{r+1}\dots y_nA_n\dots y_mA_my_{m+1}, p)$$

is of the form

$$y_1\langle A_1, p, n\rangle\dots y_r\langle A_r, p, n\rangle\dots y_n\langle A_n, p, n\rangle y_{n+1}A_{n+1}\dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}.$$

Now, one of the productions from (2.a.i), (2.b.i), and (2.c.i.A) is applicable, depending on the simulated production, m, and n:

(1) If $r + t - 1 > n$, then a production introduced by (2.a.i) is applied, so

$$\begin{aligned} &y_1\langle A_1, p, n\rangle\dots y_{r-1}\langle A_{r-1}, p, n\rangle y_r\langle A_r, p, n\rangle\\ &y_{r+1}\langle A_{r+1}, p, n\rangle\dots y_n\langle A_n, p, n\rangle y_{n+1}A_{n+1}\dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}\\ {}_{\lim}^{\ n}\!\Rightarrow_{\bar{G}}\ &y_1\langle A_1, q, n\rangle\dots y_{r-1}\langle A_{r-1}, q, n\rangle y_rx_1\langle B_1, q, n\rangle\dots x_{n-r+1}\langle B_{n-r+1}, q, n\rangle x_{n-r+2}B_{n-r+2}\dots x_tB_tx_{t+1}\\ &y_{r+1}A_{r+1}\dots y_nA_ny_{n+1}A_{n+1}\dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}.\end{aligned}$$

(2) If $r + t - 1 \le n$ and $m + t - 1 > n$, then a production introduced by (2.b.i) is applied, so

$$\begin{aligned} &y_1\langle A_1, p, n\rangle\dots y_{r-1}\langle A_{r-1}, p, n\rangle y_r\langle A_r, p, n\rangle\\ &y_{r+1}\langle A_{r+1}, p, n\rangle\dots y_{n-t+1}\langle A_{n-t+1}, p, n\rangle\\ &y_{n-t+2}\langle A_{n-t+2}, p, n\rangle\dots y_n\langle A_n, p, n\rangle y_{n+1}A_{n+1}\dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}\\ {}_{\lim}^{\ n}\!\Rightarrow_{\bar{G}}\ &y_1\langle A_1, q, n\rangle\dots y_{r-1}\langle A_{r-1}, q, n\rangle y_rx_1\langle B_1, q, n\rangle\dots x_t\langle B_t, q, n\rangle x_{t+1}\\ &y_{r+1}\langle A_{r+1}, q, n\rangle\dots y_{n-t+1}\langle A_{n-t+1}, q, n\rangle\\ &y_{n-t+2}A_{n-t+2}\dots y_nA_ny_{n+1}A_{n+1}\dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}.\end{aligned}$$

(3) If $m + t - 1 \le n$ and $t = 0$, then a production introduced by (2.c.i.A) is applied, so

$$\begin{aligned} &y_1\langle A_1, p, n\rangle\dots y_{r-1}\langle A_{r-1}, p, n\rangle y_r\langle A_r, p, n\rangle\\ &y_{r+1}\langle A_{r+1}, p, n\rangle\dots y_n\langle A_n, p, n\rangle y_{n+1}A_{n+1}\dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}\\ {}_{\lim}^{\ n}\!\Rightarrow_{\bar{G}}\ &y_1\langle A'_1, q, n-1\rangle\dots y_{r-1}\langle A'_{r-1}, q, n-1\rangle y_rx_1\\ &y_{r+1}\langle A'_{r+1}, q, n-1\rangle\dots y_n\langle A'_n, q, n-1\rangle y_{n+1}A_{n+1}\dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}.\end{aligned}$$

Recall that the last nonterminal in every sentential form of $\bar{G}$ is from $N_2 \cup N_4$. As $\langle A_n, p, n\rangle \notin N_2 \cup N_4$, there is at least one nonterminal in the sentential form following $\langle A_n, p, n\rangle$. Therefore, a production from (2.c.i.B) can be used. This production rewrites a nonterminal $A \in (V - T) \cup N_4$ in its last component. Because we generate the language of order n, $A = A_{n+1}$, so

- either

$$\begin{aligned} &y_1\langle A'_1, q, n-1\rangle\dots y_{r-1}\langle A'_{r-1}, q, n-1\rangle y_rx_1y_{r+1}\langle A'_{r+1}, q, n-1\rangle\dots y_n\langle A'_n, q, n-1\rangle\\ &y_{n+1}A_{n+1}y_{n+2}A_{n+2}\dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}\\ {}_{\lim}^{\ n}\!\Rightarrow_{\bar{G}}\ &y_1\langle A_1, q, n\rangle\dots y_{r-1}\langle A_{r-1}, q, n\rangle y_rx_1y_{r+1}\langle A_{r+1}, q, n\rangle\dots y_n\langle A_n, q, n\rangle\\ &y_{n+1}\langle A_{n+1}, q, n\rangle y_{n+2}A_{n+2}\dots y_{m-1}A_{m-1}y_m\hat{A}_my_{m+1}\end{aligned}$$

  if $A_{n+1} \in V - T$
- or

$$\begin{aligned} &y_1\langle A'_1, q, n-1\rangle\dots y_{r-1}\langle A'_{r-1}, q, n-1\rangle y_rx_1\\ &y_{r+1}\langle A'_{r+1}, q, n-1\rangle\dots y_n\langle A'_n, q, n-1\rangle y_{n+1}\hat{A}_{n+1}y_{n+2}\\ {}_{\lim}^{\ n}\!\Rightarrow_{\bar{G}}\ &y_1\langle A_1, q, n\rangle\dots y_{r-1}\langle A_{r-1}, q, n\rangle y_rx_1y_{r+1}\langle A_{r+1}, q, n\rangle\dots y_n\langle A_n, q, n\rangle y_{n+1}\langle\hat{A}_{n+1}, q, n\rangle y_{n+2}\end{aligned}$$

  if $\hat{A}_{n+1} \in N_4$.

To see that for a given sentential form and a production of G, there exists only one of the above derivations in $\bar{G}$, let us make the following observations:

(1) For a given sentential form (which determines the number of nonterminals m, first $k \le n$ nonterminals and the state p) and a production of G (which determines the new state q, $B_1, \dots, B_t$ and the constants r and t), there exists only one production in $\bar{G}$ that simulates this production.

(2) The simulating production rewrites all nonterminals from $N_1 \cup N_2$ appearing in the sentential form of $\bar{G}$. Indeed, as k is included in each of these nonterminals, no production rewriting less than k nonterminals can be used.
(3) The last nonterminal of the sentential form in $\bar{G}$ is always from $N_2 \cup N_4$.
(4) The production of G is properly simulated by the corresponding production of $\bar{G}$ — that is, all the constructed productions satisfy $A_i \notin \alpha(p)$ for every $1 \leq i < r$ and p is changed to q in all nonterminals from $N_1 \cup N_2$. In addition, in every nonterminal $N_1 \cup N_2$, k is updated.

Finally, notice that if the sentential form of $\bar{G}$ is of the form (1) or (2) as described in Claim 1, the sentential form obtained after performing a derivation step is of one of these forms as well. As the right-hand side of the production introduced in step (1) of the construction is of the form (1), every sentential form obtained during the derivation process satisfies the properties given in Claim 1. □

From Claim 1 and the derivations described in its proof, we see that $\bar{G}$ rewrites at most n first nonterminals in a sentential form and $L(G, n) = L(\bar{G}, n)$. □

Lemma 2. *$\mathscr{L}(PSC, n) \subseteq \mathscr{L}(ST, n)$ for all $n \geq 1$.*

Proof. Let $L(G, n)$ be a language of order n generated by a propagating scattered context grammar $G = (V, T, P, S)$. Set

$$N = \{\langle A, i\rangle : A \in V - T, 1 \leq i \leq n\}.$$

Further, set $K_1 = \{\langle p, i\rangle : p \in P, 0 \leq i < n\}$ and

$$K_2 = \{\langle p, i, j\rangle : p = (A_1, \ldots, A_k) \to (x_1, \ldots, x_k) \in P, 0 \leq i \leq n, 0 \leq j \leq k\}.$$

Define the state grammar

$$\bar{G} = (V \cup N \cup \{\bar{S}\}, T, K_1 \cup K_2 \cup \{p_0\}, \bar{P}, \bar{S}, p_0)$$

with $\bar{P}$ constructed as follows:

(1) For every $p = (S) \to (x) \in P$, add

$(\bar{S}, p_0) \to (S, \langle p, 0\rangle)$ to $\bar{P}$;

(2) For every $A \in V - T$, $p = (A_1, \ldots, A_k) \to (x_1, \ldots, x_k) \in P$, $0 \leq i < n$, add
 (a) $(A, \langle p, i\rangle) \to (\langle A, i + 1\rangle, \langle p, i + 1\rangle)$,
 (b) $(A, \langle p, i\rangle) \to (\langle A, i + 1\rangle, \langle p, i + 1, k\rangle)$ to $\bar{P}$;
(3) For every $p = (A_1, \ldots, A_j, \ldots, A_k) \to (x_1, \ldots, x_j, \ldots, x_k) \in P$, $q \in P$, $A \in V - T$, $1 \leq i \leq n$, $0 \leq j \leq k$, add
 (a) $(\langle A, i\rangle, \langle p, i, j\rangle) \to (A, \langle p, i - 1, j\rangle)$,
 (b) if $j \geq 1$, add

 $(\langle A_j, i\rangle, \langle p, i, j\rangle) \to (x_j, \langle p, i - 1, j - 1\rangle)$,

 (c) $(A, \langle p, 0, 0\rangle) \to (A, \langle q, 0\rangle)$ to $\bar{P}$.

As the proof of this lemma resembles the proof of $\mathscr{L}(ST) = \mathscr{L}(CS)$ given in [4], we only sketch the basic idea behind the above construction and leave a formal version of the proof to the reader.

Every derivation step of G is simulated in two phases in $\bar{G}$. In the first phase (performed by productions from (2)), G assigns a sequence number to the first m nonterminals in the sentential form, where $m \leq n$ is selected non-deterministically. The form of the constructed productions ensures that no nonterminal is skipped during this phase. The grammar $\bar{G}$ enters the second phase by a production from (2b). In the second phase, $\bar{G}$ simulates the scattered context production backwards; it starts by applying the last context-free component of the scattered context production and ends by simulating the first context-free component. The previously numbered nonterminals are processed backwards now; again, none of them can be skipped. The state of $\bar{G}$ consists of three components. First, it contains the scattered context production which is being simulated; second, it contains the position of the nonterminal within the sentential form which is being rewritten; finally, it contains the position of the context-free component within the scattered context production whose rewriting is being simulated. If the current nonterminal coincides with the left-hand side of the currently simulated context-free component, the simulation can be performed by (3b). Every nonterminal can be skipped by a production from (3a), which only removes the sequence number assigned during the first phase. Finally, when the whole scattered context production was simulated (the last component of the state equals 0) and the sequence numbers were removed from all nonterminals (the second component of the state equals 0), the simulation of the following scattered context production can be initiated by (3c). Otherwise, the simulation is unsuccessful and the derivation is blocked. □

As $\mathscr{L}(ST, n) \subseteq \mathscr{L}(PSC, n)$ and $\mathscr{L}(PSC, n) \subseteq \mathscr{L}(ST, n)$ for all $n \geq 1$, we obtain the the main result of this paper.

Theorem 1. *$\mathscr{L}(PSC, n) = \mathscr{L}(ST, n)$, for all $n \geq 1$.* □

Next, we reformulate Theorem 1 in terms of $\mathscr{L}(PUSC, n)$.

Theorem 2. *$\mathscr{L}(PUSC, n) = \mathscr{L}(ST, n)$ for all $n \geq 1$.*

Proof. To prove this theorem, we have to show that $\mathscr{L}(ST, n) \subseteq \mathscr{L}(PUSC, n)$ and $\mathscr{L}(PUSC, n) \subseteq \mathscr{L}(ST, n)$ for all $n \geq 1$.

The first inclusion can be proved similarly to the proof of Lemma 1. However, in this case, the constructed grammar must also record the sequence number of each of the first n nonterminals in the sentential form in order to ensure that the right component of the unordered scattered context production is used; otherwise, any permutation of the unordered scattered context production could be applied. Therefore, the elements of the sets N_1, N_2, and N_3 have to be changed to contain one more component which determines the order of the nonterminal in the sentential form. In addition, each of the constructed productions has to be changed so that it preserves the correct sequence numbers of these nonterminals in every sentential form. This can be easily accomplished because each of these productions rewrites all the nonterminals from $N_1 \cup N_2 \cup N_3$ in the sentential form. The rest of the proof is analogous to the proof of Lemma 1 and, therefore, left to the reader.

The second inclusion, $\mathscr{L}(PUSC, n) \subseteq \mathscr{L}(ST, n)$, can be proved as follows. For any propagating unordered scattered context grammar of order n, G, it is possible to construct a propagating scattered context grammar of the same order, $\bar{G}$, such that $L(G, n) = L(\bar{G}, n)$. Indeed, if we construct $\bar{G}$ so that it contains all possible permutations of every production of G, we obtain a $\bar{G}$ satisfying these properties. Therefore, $L(G, n) = L(\bar{G}, n)$ and $\mathscr{L}(PUSC, n) \subseteq \mathscr{L}(PSC, n)$. As $\mathscr{L}(PSC, n) = \mathscr{L}(ST, n)$, we obtain $\mathscr{L}(PUSC, n) \subseteq \mathscr{L}(ST, n)$. $\square$

Recall that $\mathscr{L}(CF) = \mathscr{L}(ST, 1) \subset \mathscr{L}(ST, 2) \subset \cdots \subset \mathscr{L}(ST, \infty) \subset \mathscr{L}(ST) = \mathscr{L}(CS)$, where every $\mathscr{L}(ST, n)$, for $n \geq 1$, is an abstract family of languages (see [4]). These properties together with Theorems 1 and 2 imply the following two corollaries.

Corollary 1.

$$\begin{aligned} & \mathscr{L}(PUSC, 1) &&= \mathscr{L}(PSC, 1) &&= \mathscr{L}(ST, 1) &&= \mathscr{L}(CF) \\ \subset\ & \mathscr{L}(PUSC, 2) &&= \mathscr{L}(PSC, 2) &&= \mathscr{L}(ST, 2) \\ & \vdots \\ \subset\ & \mathscr{L}(PUSC, \infty) &&= \mathscr{L}(PSC, \infty) &&= \mathscr{L}(ST, \infty) \\ \subset\ & \mathscr{L}(CS). \quad \square \end{aligned}$$

Corollary 2. *Every $\mathscr{L}(PSC, n)$ and $\mathscr{L}(PUSC, n)$, where $n \geq 1$, is an abstract family of languages—that is, this family is closed under the operations of union, concatenation, Kleene plus, inverse homomorphism, ε-free homomorphism, and intersection with a regular language.* $\square$

4. Conclusion

We have demonstrated that limiting derivations performed by scattered context grammars and unordered scattered context grammars to the first n nonterminals gives rise to an infinite hierarchy of languages. This result is of some practical interest in terms of compilers (see [8]). Indeed, when constructing a compiler based on a grammatical model, we usually need to restrict this model in order to make the compiler more effective. The presented result shows that if the model is based on scattered context grammars, by limiting the width of the window in which the context dependency is checked, we also limit the power of the resulting compiler. In certain situations, such as parsing of streamed data, limiting the context dependency check to a finite window is necessary because the exact length of the input is unknown.

From a theoretical point of view, the achieved results are interesting too. It is well known that $\mathscr{L}(PUSC) \subset \mathscr{L}(PSC)$. However, when using n-limited derivations, $\mathscr{L}(PUSC, n) = \mathscr{L}(PSC, n)$. On the other hand, the definition of n-limited derivations induces the following problem: is it possible to construct a scattered context grammar which checks whether it rewrites the first n nonterminals without any additional explicit restriction imposed on its derivations (that is to define a scattered context grammar of order n in the way analogous to a state grammar of order n) and obtain the same results? How would this modification influence the generative power of unordered scattered context grammars? We propose these open problems for further study.

Acknowledgements

The first author was supported by GAČR grant 201/07/0005 and Research Plan MSM 021630528 and the second author was supported by GAČR grant 102/05/H050.

References

[1] H. Fernau, Scattered context grammars with regulation, Annals of Bucharest University, Mathematics-Informatics Series 45 (1) (1996) 41–49.
[2] S. Greibach, J. Hopcroft, Scattered context grammars, Journal of Computer and System Sciences 3 (1969) 233–247.
[3] D. Hauschildt, M. Jantzen, Petri net algorithms in the theory of matrix grammars, Acta Informatica 31 (8) (1994) 719–728.
[4] T. Kasai, An hierarchy between context-free and context-sensitive languages, Journal of Computer and System Sciences 4 (5) (1970) 492–508.
[5] T. Masopust, J. Techet, Leftmost derivations of propagating scattered context grammars: A new proof, Discrete Mathematics and Theoretical Computer Science 10 (2) (2008) 39–46.
[6] O. Mayer, Some restrictive devices for context-free grammars, Information and Control 20 (1972) 69–92.
[7] A. Meduna, Automata and Languages: Theory and Applications, Springer, 2000.

[8] A. Meduna, Elements of Compiler Design, Taylor and Francis, 2008.
[9] A. Meduna, J. Techet, Maximal and minimal scattered context rewriting, in: FCT 2007 Proceedings, vol. 4639, Springer Verlag, Budapest, 2007, pp. 412–423.
[10] D. Milgram, A. Rosenfeld, A note on scattered context grammars, Information Processing Letters 1 (1971) 47–50.
[11] A. Salomaa, Formal Languages, Academic Press, 1973.