CD Grammar Systems with Two Propagating Scattered Context Components Characterize the Family of Context Sensitive Languages

Jakub Martiško

Centre of Excellence IT4Innovations, Department of Information Systems, Faculty of Information Technology, Brno University of Technology
Božetěchova 2, Brno 612 66, Czech republic
imartisko@fit.vutbr.cz

Alexander Meduna

Centre of Excellence IT4Innovations, Department of Information Systems, Faculty of Information Technology,
Brno University of Technology
Božetěchova 2, Brno 612 66, Czech republic
meduna@fit.vutbr.cz

Zbyněk Křivka

Centre of Excellence IT4Innovations, Department of Information Systems, Faculty of Information Technology,
Brno University of Technology
Božetěchova 2, Brno 612 66, Czech republic
krivka@fit.vutbr.cz

Received (Day Month Year)
Accepted (Day Month Year)
Communicated by (xxxxxxxxxx)

The $\mathscr{L}(PSCG) = \mathscr{L}(CS)$ problem asks whether propagating scattered context grammars and context sensitive grammars are equivalent. The presented paper reformulates and answers this problem in terms of CD grammar systems. More specifically, it characterizes the family of context sensitive languages by two-component CD grammar systems with propagating scattered context rules.

1. Introduction

Are propagating scattered context grammars as powerful as context sensitive grammars? This question customarily referred to as the $\mathscr{L}(PSCG) = \mathscr{L}(CS)$ (see [5]) problem, represents a long standing open problem in formal language theory.

To address this hard open problem as close as possible, there were attempts to introduce subtle modifications of propagating scattered context grammars together with a demonstration of their generative power in the comparison with context sensitive grammars (see [8]). For instance, Gonczarowski and Warmuth (see [4]) introduced extended propagating scattered context grammars and propagating unordered scattered context grammars such that the former generates context sensitive languages and the latter is even less powerful than propagating scattered context grammars.

The present paper reformulates this problem and answers it in terms of CD grammar systems. More precisely, the paper introduces CD grammar systems whose components are propagating scattered context grammars. Then, it demonstrates that two-component grammar systems of this kind generate the family of context-sensitive languages, thus the answer to this problem is in affirmation if the problem is reformulated in the above way. In the end of the paper, we study the descriptional complexity of components of such CD grammar systems in order to show that the maximum context sensitivity of all components can be reduced to one or zero.

2. Preliminaries

We assume that the reader is familiar with formal language theory (see [7, 3, 9, 10] for details). For an alphabet (finite nonempty set) V, V^* represents the free monoid generated by V under the operation of concatenation. The unit of V^* is denoted by ε. Let $n \geq 0$ and $x_1, x_2, \ldots, x_n \in V$. The length of string $x_1 \cdots x_n \in V^*$ is denoted as $|x_1 \cdots x_n|$ and is equal to n. Similarly by $|x_1 \cdots x_n|_N$, the length of string when counting only symbols of N is denoted. The function $\mathrm{alph}(x_1 \cdots x_n)$ is defined as $\mathrm{alph}(\alpha) = \{x : \alpha = \beta x \gamma, x \in V\}$.

A *scattered context grammar (SCG)* is a quadruple $G = (N, T, P, S)$, where N and T are alphabets of nonterminal and terminal symbols respectively, where $N \cap T = \emptyset$, further let $V = N \cup T$. $S \in N$ is the starting symbol. P is a nonempty finite set of rules of the form $(A_1, \ldots, A_n) \to (\alpha_1, \ldots \alpha_n)$, where $A_i \in N$, $\alpha_i \in V^*, 1 \leq i \leq n$, for some $n \geq 1$. Let $u, v \in V^*$, where $u = u_1 A_1 u_2 A_2 u_3 \ldots u_n A_n u_{n+1}$ and $v = u_1 \alpha_1 u_2 \alpha_2 u_3 \ldots u_n \alpha_n u_{n+1}$, $(A_1, A_2, A_3, \ldots, A_n) \to (\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_n) \in P$, where $u_i \in V^*$ for all $1 \leq i \leq n+1$; then $u \Rightarrow v$ in G. For each rule $r \in P$, $\mathrm{len}(r)$ denotes the number of components of r. The *maximum context sensitivity*, denoted by $\mathrm{mcs}(G)$, is defined as $\mathrm{mcs}(G) = \max(\{\mathrm{len}(r) - 1 : r \in P\})$.

The language generated by SCG G is defined as $L(G) = \{x : S \Rightarrow^* x, x \in T^*\}$, where $\Rightarrow^*$ and $\Rightarrow^+$ denote the transitive-reflexive closure and the transitive closure of $\Rightarrow$, respectively. A SCG is said to be *propagating (PSCG)* iff each $(A_1, \ldots, A_n) \to (\alpha_1, \ldots \alpha_n) \in P$ satisfies $\alpha_i \neq \varepsilon, 1 \leq i \leq n$. $\mathscr{L}(SCG)$ and $\mathscr{L}(PSCG)$ denote the families of languages generated by SCGs and PSCGs, respectively.

A *context-sensitive grammar (CSG)* is a quadruple $G = (N, T, P, S)$, where N, T, V, and S has the same meaning as in SCGs. P is a nonempty finite set of rules of the form $\alpha \to \beta$, where $|\alpha|_N \geq 1$, $\alpha, \beta \in V^*$ and each rule $\alpha \to \beta \in P$ satisfies $|\alpha| \leq |\beta|$. Let $u, v \in V^*$, $u = u_1 \alpha u_2$, $v = u_1 \beta u_2$, $\alpha \to \beta \in P$ where $u_1, u_2, \alpha, \beta \in V^*$, $|\alpha|_N \geq 1$, then $u \Rightarrow v$ in G.

The language generated by CSG G is defined as $L(G) = \{x : S \Rightarrow^* x,\ x \in T^*\}$, where $\Rightarrow^*$ and $\Rightarrow^+$ denote the transitive-reflexive closure and the transitive closure of $\Rightarrow$, respectively. The family of languages generated by CSGs is denoted by $\mathscr{L}(CS)$.

A grammar $G = (N, T, P, S)$ is in *Kuroda normal form* if every rule in P has one of the following forms:

(1) $AB \to CD$
(2) $A \to CD$
(3) $A \to C$
(4) $A \to a$

where $A, B, C, D \in N$ and $a \in T$. Recall that every CSG can be transformed into an equivalent grammar in Kuroda normal form (see [7, 6]). Notice that we do not require more restrictive variant of Kuroda normal form without the rules of the form $A \to C$.

Without any loss of generality, for any CSG we assume their equivalent in the Kuroda normal form in what follows.

To emphasize that rule p was used during the derivation step, we sometimes write $\alpha \Rightarrow \beta[p]$.

A *cooperating distributed grammar system (CDGS)* (see [1, 9, 2]) of degree n is n+3 tuple $G = (N, T, S, P_1, P_2, \dots, P_n)$, where N, T, V, and S has the same meaning as in SCGs. $P_i, 1 \leq i \leq n$ are nonempty finite sets (called components) of rewriting rules over V. For a CDGS $G = (N, T, S, P_1, P_2, \dots, P_n)$, the terminating ($t$) derivation by the i-th component, denoted as $\Rightarrow^t_{P_i}$ is defined as $u \Rightarrow^t_{P_i} v$ iff $u \Rightarrow^*_{P_i} v$ and there is no $z \in V^*$ such that $v \Rightarrow_{P_i} z$. The language generated by CDGS $G = (N, T, S, P_1, P_2, \dots, P_n)$ working in t mode is defined as $L(G) = \{x : S \Rightarrow^t_{P_{i_1}} x_1 \Rightarrow^t_{P_{i_2}} x_2 \dots \Rightarrow^t_{P_{i_m}} x,\ m \geq 1,\ 1 \leq i_j \leq n,\ 1 \leq j \leq m,\ x \in T^*\}$.

In this paper, CDGS with propagating scattered context rules (SCGS) and CDGS with context-sensitive rules are considered.

3. Main Results

In this section, the identity of $\mathscr{L}(SCGS)$ and $\mathscr{L}(CS)$ is demonstrated.

Lemma 1. $\mathscr{L}(SCGS) \subseteq \mathscr{L}(CS)$

Proof. Recall that [3] shows that any scattered context grammar can be simulated by context-sensitive grammar. Similarly, [2] shows that any CDGS with context-sensitive components working in t mode can be transformed to an equivalent CSG. Based on those two facts, it is easy to show that any SCGS can be simulated by a CSG. □

Lemma 2. $\mathscr{L}(CS) \subseteq \mathscr{L}(SCGS)$

Take any CSG $G = (N, T, P, S)$ satisfying Kuroda normal form. An equivalent SCGS $\Gamma = (N_{GS}, T, {}^{\triangle}S, P_1, P_2)$ can be constructed using the following constructions. Set $N_{GS} = N \cup \{!\} \cup N_T \cup N_{first} \cup N_{CF} \cup N_{CS} \cup N_{cur}$ $(! \notin N \cup T)$. Where:

$$
\begin{aligned}
N_T &= \{a^{'} : \forall a \in T\} \\
N_{\triangle} &= \{{}^{\triangle}X : \forall X \in N \cup N_T\} \\
N_{\triangledown} &= \{{}^{\triangledown}X : \forall X \in N \cup N_T\} \\
N_{\diamond} &= \{{}^{\diamond}X : \forall X \in N \cup N_T\} \\
N_{first} &= N_{\triangle} \cup N_{\triangledown} \cup N_{\diamond} \\
N_{CF\triangle} &= \{X_{|} : \forall X \in N_{\triangle}\} \\
N_{CF\triangledown} &= \{X_{|} : \forall X \in N_{\triangledown}\} \\
N_{CF\diamond} &= \{X_{|} : \forall X \in N_{\diamond}\} \\
N_{CF} &= \{{}_{|}X_{|} : \forall X \in N\} \cup N_{CF\triangle} \cup N_{CF\triangledown} \cup N_{CF\diamond} \\
N_{CS} &= \{{}_{|}X_{<} : \forall X \in N\} \cup \{X_{<} : \forall X \in N_{first}\} \cup \{{}_{>}X_{|} : \forall X \in N\} \\
N_{cur} &= \{X_{<}^{\wedge} : \forall X \in N \cup N_{first}\} \cup \{X_{|}^{\wedge} : \forall X \in N \cup N_{first}\}.
\end{aligned}
$$

Analogically to sets $N_{CF\triangle}, N_{CF\triangledown}$ and $N_{CF\diamond}$, we call subsets of N_{CS} and N_{cur} constructed using the set $N_{\triangle}$ as $N_{CS\triangle}$, and $N_{cur\triangle}$, respectively. We use similar naming convention for subsets constructed using the $N_{\triangledown}$ and $N_{\diamond}$.

Set P_1 to the union of the following sets:

$$
\begin{aligned}
P_T^1 = &\{({}^{\triangle}X) \to ({}^{\diamond}X) : \forall X \in N \cup N_T\} \\
&\cup \{({}^{\diamond}X, a^{'}) \to ({}^{\diamond}X, a) : \\
&\quad \forall X \in N \cup N_T, \forall\, a^{'} \in N_T\} \\
&\cup \{({}^{\diamond}a^{'}) \to (a) : \forall\, a^{'} \in N_T\} \\
P_{AtoBC}^1 = &\{({}^{\triangle}X, A) \to ({}^{\triangledown}X_{|}, {}_{|}B_{|}\,{}_{|}C_{|}) : \\
&\quad \forall X \in N_T \cup N, \forall p \in P, p = A \to BC\} \\
&\cup \{({}^{\triangle}A) \to ({}^{\triangledown}B_{|}\,{}_{|}C_{|}) : \forall p \in P, p = A \to BC\} \\
P_{AtoB}^1 = &\{({}^{\triangle}X, A) \to ({}^{\triangledown}X_{|}, {}_{|}B_{|}) : \\
&\quad \forall X \in N_T \cup N, \forall p \in P, p = A \to B\} \\
&\cup \{({}^{\triangle}A) \to ({}^{\triangledown}B_{|}) : \forall p \in P, p = A \to B\}
\end{aligned}
$$

$$\begin{aligned}
P^1_{Atoa} = &\{(^{\triangle}X, A) \rightarrow (^{\triangledown}X_{|}, {}_{|}a'_{|}) : \\
&\forall X \in N_T \cup N, \forall p \in P, p = A \rightarrow a\} \\
&\cup \{(^{\triangle}A) \rightarrow (^{\triangledown}a'_{|}) : \forall p \in P, p = A \rightarrow a\} \\
P^1_{ABtoCD} = &\{(^{\triangle}X, A, B) \rightarrow (^{\triangledown}X_{|}, {}_{|}C_{<}, {}_{>}D_{|}) : \\
&\forall X \in N_T \cup N, \forall p \in P, p = AB \rightarrow CD\} \\
&\cup \{(^{\triangle}A, B) \rightarrow (^{\triangledown}C_{<}, {}_{>}D_{|}) : \\
&\forall p \in P, p = AB \rightarrow CD\} \\
P^1_{phase2} = &\{(X, B) \rightarrow (X, {}_{|}B_{|}) : \\
&\forall B \in N \cup N_T, X \in N_{CS\triangledown} \cup N_{CF\triangledown}\}
\end{aligned}$$

Set P_2 to the union of these subsets:

$$\begin{aligned}
P^2_{init} = &\{(^{\triangledown}X_{|}) \rightarrow (^{\triangle}X^{\wedge}_{|}) : \forall X \in N_T \cup N\} \\
&\cup \{(^{\triangledown}X_{<}) \rightarrow (^{\triangle}X^{\wedge}_{<}) : \forall X \in N_T \cup N\} \\
P^2_{check} = &\{(A^{\wedge}_{|}, {}_{|}B_{|}) \rightarrow (A, B^{\wedge}_{|}) : \forall X, A, B \in N_T \cup N\} \\
&\cup \{(A^{\wedge}_{<}, {}_{>}B_{|}) \rightarrow (A, B^{\wedge}_{|}) : \forall X, A, B \in N_T \cup N\} \\
P^2_{checkf} = &\{(^{\triangle}A^{\wedge}_{|}, {}_{|}B_{|}) \rightarrow (^{\triangle}A, B^{\wedge}_{|}) : \forall A, B \in N_T \cup N\} \\
&\cup \{(^{\triangle}A^{\wedge}_{<}, {}_{>}B_{|}) \rightarrow (^{\triangle}A, B^{\wedge}_{|}) : \forall A, B \in N_T \cup N\} \\
P^2_{end} = &\{(^{\triangle}A^{\wedge}) \rightarrow (^{\triangle}A) : \forall A \in N_T \cup N\} \\
&\cup \{(^{\triangle}A, B^{\wedge}_{|}) \rightarrow (^{\triangle}A, B) : \forall A, B \in N_T \cup N\} \\
&\cup \{(^{\triangle}A^{\wedge}_{|}) \rightarrow (^{\triangle}A) : \forall A \in N_T \cup N\} \\
P^2_{block} = &\{({}_{|}X_{|}) \rightarrow (!) : \forall X \in N_T \cup N\}
\end{aligned}$$

Basic Idea 1. *Now, we briefly describe how the resulting SCGS* Γ *simulates the input CSG* G*. The system consists of two components, both working in t mode. The computation of* Γ *consists of two phases. During the first one, all terminals are represented by a nonterminal variant of themselves. The simulation itself takes place during the first phase.*

The simulation in Γ *of each application of one rule of* G *consists of two parts. Firstly, the first component applies the selected rule using the modified nonterminals contained in* N_{CS} *and* N_{CF}*. Symbols of the type* ${}_{|}X_{<}$ *denote that the rewriting is done in a context-sensitive way and that the remaining symbol on the right-hand side of the rule should appear immediately right of the symbol. Similarly* ${}_{>}X_{|}$ *denotes that the rest of the right-hand side of the rule should appear immediately left of the symbol. Symbols of the form* ${}_{|}X_{|}$ *then represent context-free rewriting. After the application of the rule, the first component rewrites all remaining symbols to* *their context-free variant and then deactivates. This is done using the rules of the set* P^1_{phase2}*. The fact that only one of the rules was applied is checked using the first symbol of the sentential form. This symbol is of the form* ${}^{\triangle}X$ *or* ${}^{\triangledown}X$ *(plus the context-sensitive and context-free versions), where the marks* $\triangle$ *and* $\triangledown$ *indicate, whether next rule should be simulated, or remaining symbols should be rewritten to their context-free variant.*

The second component then checks, whether the first component applied the rule correctly. This is done using the special $\wedge$ *mark. This symbol indicates, which symbol is currently checked, we call this symbol* current symbol*. Symbols are checked in pairs, where the first symbol of the pair is the current symbol and the second symbol is some symbol right of the first one. During this check, the special marks* $(|, <, >)$ *on the adjacent sides of those symbols are checked and removed and the* $\wedge$ *mark is moved to the other symbol of the pair. Since the first symbol of the pair is always the current symbol, the* $\wedge$ *moves from the left side of the sentential form to the right, with no way of returning back left. When all of the symbols are checked, the second component is deactivated and the first one simulates new rule. Since the components have scattered context rules, it is not guaranteed that adjacent symbols are always checked by the second component. Because of this, set of rules* P^2_{block} *is created. When some of the symbols is skipped during the checking phase, these rules will block the generation of sentence by* Γ*.*

The second phase, which rewrites all nonterminals to terminals, is started by rewriting of the first symbol ${}^{\triangle}X$ *to* ${}^{\diamond}X$*. Then for each symbol* $a^{'}$*, there is a rule of the form* $({}^{\diamond}X, a^{'}) \to ({}^{\diamond}X, a)$*, where* a *is corresponding terminal symbol. Finally, the leftmost symbol itself is rewritten to its terminal form. Since all the rules of all components always check the first symbol, after this step no further rewriting can be done and all nonterminals that remain in the sentential form cannot be removed. This phase is represented by set* P^1_T*.*

Next, we sketch a formal proof that $L(G) = L(\Gamma)$. Its fully rigorous version is left to the reader.

Claim 1. *In any sentential form, there is always at most one symbol marked with any of* $\triangle, \triangledown, \diamond$*.*

Proof. Observe that no rule contains more than one symbol marked with any of $\diamond, \triangledown, \triangle$ on the right-hand side. Furthermore observe that if any marked symbol does appear on the right-hand side of a rule, there is also a marked symbol on the left-hand side of the same rule. Thus no new marked symbols can be introduced into the sentential form. □

Claim 2. *Any derivation that generates a sentence ends with a sequence of rules of the form* $p_1 p_{2_1} \dots p_{2_n} p_3$*, where* $p_1, p_{2_i}, p_3 \in P^1_T, 1 \leq i \leq n, n \geq 0$*, where* p_1*,* p_{2_i} *and* p_3 *are from the first, second and third subset of* P^1_T*, respectively. No rule from* P^1_T *is applied before this sequence.*

Proof. Recall that only rules which have terminals on the right-hand side are in set P_T^1 which is defined as follows (in this proof, we named each of its subsets for the sake of simplicity):

$$\begin{aligned} P_T^1 &= {}_1P_T^1 \cup {}_2P_T^1 \cup {}_3P_T^1 \\ {}_1P_T^1 &= \{({}^{\triangle}X) \to ({}^{\diamond}X) : \forall X \in N \cup N_T\} \\ {}_2P_T^1 &= \{({}^{\diamond}X, a^{'}) \to ({}^{\diamond}X, a) : \forall X \in N \cup N_T, \forall\, a^{'} \in N_T\} \\ {}_3P_T^1 &= \{({}^{\diamond}a^{'}) \to (a) : \forall\, a^{'} \in N_T\} \end{aligned}$$

Suppose any sentential form χ such that $\chi = {}^{\triangle}x_0^{'}\, x^{'}{}_1 \dots x^{'}{}_n$, where $x^{'}{}_i \in N_T$ $0 \leq i \leq n$ and ${}^{\triangle}x_0^{'} \in N_{\diamond}$. Observe that all rules that do rewriting to terminals check the existence of a symbol ${}^{\diamond}X$ in the sentential form. This symbol is created in a following way:

$${}^{\triangle}x_0^{'}\, x^{'}{}_1 \dots x^{'}{}_n \Rightarrow {}^{\diamond}x_0^{'}\, x^{'}{}_1 \dots x^{'}{}_n[p], p \in {}_1P_T^1$$

Careful examination of sets P_1 and P_2 shows that only rules with ${}^{\diamond}x_0^{'}$ on its left-hand side are in sets ${}_2P_T^1$ and ${}_3P_T^1$. Suppose the following derivation

$${}^{\diamond}x_0^{'}\, x^{'}{}_1 \dots x^{'}{}_n \Rightarrow x_0\, x^{'}{}_1 \dots x^{'}{}_n[p], p \in {}_3P_T^1, x_0 \in T$$

Based on Claim 1, $\mathrm{alph}(x^{'}{}_1 \dots x^{'}{}_n) \cap (N_{\triangle} \cup N_{\triangledown} \cup N_{\diamond}) = \emptyset$. Each rule $p \in P_1$ contains some symbol from $N_{\triangle} \cup N_{\triangledown} \cup N_{\diamond}$ on its left-hand side. There is thus no $\chi^{'} \neq \chi, \chi = x_0\, x^{'}{}_1 \dots x^{'}{}_n$ such that $\chi \Rightarrow \chi^{'}[p], p \in P_1$. Further no rule from P_2 can be used (see claim 5). For any successful derivation $p \in {}_3P_T^1$ must thus be used as a last rule of this derivation.

Suppose ${}^{\triangle}x_0^{'}\, x^{'}{}_1 \dots x^{'}{}_n \Rightarrow {}^{\diamond}x_0^{'}\, x^{'}{}_1 \dots x^{'}{}_n[p], p \in {}_1P_T^1$, and further let $\chi_1 = {}^{\diamond}x_0^{'}\, x^{'}{}_1 \dots x^{'}{}_n$ where $|\chi_1| > 1$. Based on the previous paragraph, in any successful derivation, the following sequence of rules has to be applied

$$\chi_1 \Rightarrow \chi_2[p_1] \Rightarrow \dots \Rightarrow \chi_n[p_n]$$

where $\chi_n = {}^{\diamond}x_0^{'}x_1 \dots x_n$, $p_i \in {}_2P_T^1$, $1 \leq i \leq n$. For each χ_i and χ_{i+1} with $1 \leq i \leq n-1$, $|\chi_i|_T = |\chi_{i+1}|_T - 1$. □

We have just shown that the rules from P_T^1 are only applied right before the end of the successful simulation. Consequently, we do not mention this subset in any of the following proofs.

Claim 3. *The first component of* Γ *rewrites sentential forms of the form* ${}^{\triangle}X\alpha$ *to a string of one of the following forms*

(1) ${}^{\triangledown}Y_{|}\beta$
(2) ${}^{\triangledown}Y_{<}\gamma$

where $X, Y \in N \cup N_T$, $\alpha \in (N \cup N_T)^*$, $\beta, \gamma \in (N_{CS} \cup N_{CF})^*$ *(such that Claim 1 holds) where either (a) or (b) given next is true:*

(a) $\beta \in (N_{CF})^*$
(b) $\beta = Y_0 \dots {}_{|}U_{<} \dots {}_{>}V_{|} \dots Y_n$, *where* $Y_i \in N_{CF}, 0 \leq i \leq n$ *and* ${}_{|}U_{<}, {}_{>}V_{|} \in N_{CS}$

and $\gamma = Y_0 \dots {}_{>}V_{|} \dots Y_n$, *where* $Y_i \in N_{CF}, 0 \leq i \leq n$ *and* ${}_{>}V_{|} \in N_{CS}$.

Proof. Consider sentential form ${}^{\triangle}X\alpha$ defined as above, where $\alpha = X_0 \dots X_n$. Since there is no symbol from the alphabet $N_{cur} \cup N_{\nabla}$ only rules of the first component can be used.

From ${}^{\triangle}X\alpha$, Γ makes a derivation step in one of the following eight ways (each derivation corresponds to one subset of the rules of the first component of Γ):

(1) ${}^{\triangle}X X_0 \dots X_{i-1} A X_{i+1} \dots X_n$
$\Rightarrow {}^{\nabla}X_{|} X_0 \dots X_{i-1} {}_{|}B_{|} {}_{|}C_{|} X_{i+1} \dots X_n$
(2) ${}^{\triangle}A X_0 \dots X_n$
$\Rightarrow {}^{\nabla}B_{|} {}_{|}C_{|} X_0 \dots X_n$
(3) ${}^{\triangle}X X_0 \dots X_{i-1} A X_{i+1} \dots X_n$
$\Rightarrow {}^{\nabla}X_{|} X_0 \dots X_{i-1} {}_{|}B_{|} X_{i+1} \dots X_n$
(4) ${}^{\triangle}A X_0 \dots X_n$
$\Rightarrow {}^{\nabla}B_{|} X_0 \dots X_n$
(5) ${}^{\triangle}X X_0 \dots X_{i-1} A X_{i+1} \dots X_n$
$\Rightarrow {}^{\nabla}X_{|} X_0 \dots X_{i-1} {}_{|}a'_{|} X_{i+1} \dots X_n$
(6) ${}^{\triangle}A X_0 \dots X_n$
$\Rightarrow {}^{\nabla}a'_{|} X_0 \dots X_n$
(7) ${}^{\triangle}X X_0 \dots X_{i-1} A\delta B X_{i+1} \dots X_n$
$\Rightarrow {}^{\nabla}X_{|} X_0 \dots X_{i-1} {}_{|}C_{<} \delta {}_{>}D_{|} X_{i+1} \dots X_n$
(8) ${}^{\triangle}A\delta B X_0 \dots X_n$
$\Rightarrow {}^{\nabla}C_{<} \delta {}_{>}D_{|} X_0 \dots X_n$

where $\delta \in (N \cup N_T)^*$. Observe that each of the generated strings is in one of the following forms:

- ${}^{\nabla}X_{|} \delta_1 B \delta_2 C \delta_3$ (1-7)
- ${}^{\nabla}X_{<} \delta_1 {}_{>}D_{|} \delta_2$ (8)

where $\delta_1, \delta_2, \delta_3 \in (N \cup N_T)^*$, $D \in N_{CS}$ (the second subset) and either $B, C \in N_{CF}$ or $B \in N_{CS}$ (the first subset) and $C \in N_{CS}$ (the second subset).

After this first rule is applied, the sentential form contains symbol marked with ∇. Since the components of Γ work in t mode, rules of the first component have to be applied as long as there are some symbols that can be rewritten. This means that the rules from the set P^1_{phase2} have to be used now. Because $\delta_1, \delta_2, \delta_3 \in (N \cup N_T)^*$ and the left-hand sides of the rules from P^1_{phase2} are defined for all symbols in $N \cup N_T$. Substring $\delta_1 = Z_0 \dots Z_n$, $Z_i \in N \cup N_T$ is rewritten to $\delta_1' = {}_{|}Z_{0|} \dots {}_{|}Z_{n|}$,

$Z_i \in N \cup N_T, 0 \leq i \leq n$. The same applies to δ_2, δ_3. By using P^1_{phase2} we obtain one of the following sentential forms:

$$\begin{aligned} &{}^{\triangledown}X_{|}\delta_1 B\delta_2 C\delta_3 \Rightarrow^* {}^{\triangledown}Y_{|}\beta \\ &{}^{\triangledown}X_{<}\delta_1\,{}_{>}D_{|}\delta_2 \Rightarrow^* {}^{\triangledown}Y_{<}\gamma \end{aligned}$$

□

Claim 4. *During its activation, the first component applies no more than one rule of the simulated CSG. This follows from Claim 3 and its proof.*

Claim 5. *The second component of* Γ *rewrites any sentential form of the form* ${}^{\triangledown}X_{|}\,{}_{|}X_{0|}\ldots{}_{|}X_{n|}$ *to a string of the form* ${}^{\triangle}X X_1 \ldots X_n$*, where* $X_i \in N \cup N_T, 0 \leq i \leq n$ *.*

Proof. Suppose sentential form[a] $\chi = {}^{\triangledown}X_{|}\,{}_{|}X_{0|}\ldots{}_{|}X_{n|}$ where $X_i \in N \cup N_T, 0 \leq i \leq n$. Observe that $|\chi|_{N_{cur}} = 0$. Only rules[b] that can be used are thus from the first subset of P^2_{init}. This leads to

$$\chi = {}^{\triangledown}X_{|}\,{}_{|}X_{0|}\ldots{}_{|}X_{n|} \Rightarrow \chi_0 = {}^{\triangle}X^{\wedge}_{|}\,{}_{|}X_{0|}\ldots{}_{|}X_{n|}$$

The only rule applicable to χ_0 must be from the set P^2_{checkf}. This leads to:

$$\chi_0 = {}^{\triangle}X^{\wedge}_{|}\,{}_{|}X_{0|}\ldots{}_{|}X_{n|} \Rightarrow \chi_1 = {}^{\triangle}X\alpha_1\, X^{\wedge}_{i_1|}\alpha_2$$

where $\alpha_1, \alpha_2 \in N^*_{CF}$. Again, careful observation of rules of the set P_2 shows that only rules from the set P^2_{check} and P^2_{end} may be used. The first option leads to following derivations:

$${}^{\triangle}X\alpha^1_1\, X^{\wedge}_{i_1|}\alpha^2_1 \Rightarrow {}^{\triangle}X\alpha^1_1 X_{i_1}\alpha^1_2\, X^{\wedge}_{i_2|}\alpha^2_2 \Rightarrow \cdots \Rightarrow {}^{\triangle}X\alpha^1_1 X_{i_1}\alpha^1_2 X_{i_2}\ldots\alpha^1_n\, X^{\wedge}_{i_n|}\alpha^2_n$$

where $\alpha^j_k \in N^*_{CF}, 1 \leq k \leq n, j \in \{1,2\}$. Further, there is no rule p such that:

$${}^{\triangle}X\alpha^1_1 X_{i_1}\alpha^1_2 X_{i_2}\ldots\alpha^1_n\, X^{\wedge}_{i_n|}\alpha^2_n \Rightarrow {}^{\triangle}X\alpha^1_1 X_{i_1}\alpha^1_2 X_{i_2}\ldots Y^{\wedge}_{|}\ldots\alpha^1_n X_{i_n}\alpha^2_n[p]$$

Suppose ${}^{\triangle}X\alpha^1_1 X_{i_1}\alpha^1_2 X_{i_2}\ldots\alpha^1_n\, X^{\wedge}_{i_n|}\alpha^2_n$ and rule $p \in P^2_{end}$:

$${}^{\triangle}X\alpha^1_1 X_{i_1}\alpha^1_2 X_{i_2}\ldots\alpha^1_n\, X^{\wedge}_{i_n|}\alpha^2_n \Rightarrow {}^{\triangle}X\alpha^1_1 X_{i_1}\alpha^1_2 X_{i_2}\ldots\alpha^1_n X i_n\alpha^2_n[p]$$

Suppose that adjacent symbols were always rewritten during the application of rules from the sets P^2_{checkf} and P^2_{check}. This would mean that $\alpha^j_k = \varepsilon, 1 \leq k \leq n, j \in \{1,2\}$ and we would thus obtain the desire sentential form ${}^{\triangle}X X_1 \ldots X_n$, where $X_i \in N \cup N_T, 0 \leq i \leq n$.

If, on the other hand, there was some $\alpha^m_l \neq \varepsilon, 1 \leq l \leq n, m \in 1,2$ this would mean that

$${}^{\triangle}X^{\wedge}_{|}\,{}_{|}X_{0|}\ldots{}_{|}X_{n|} \Rightarrow^* {}^{\triangle}X\alpha$$

[a]The case where $|\chi|$=1 is trivial and is left to the reader.
[b]We ignore the set of blocking rules P^2_{block} for now.

where $|\alpha|_{N_{cur}} = 0$ and $|\alpha|_{N_{CF}} > 0$. Since both GS components work in the t-mode, and there is some symbol from N_{CF}, the blocking symbols have to be introduced by the rules of the P^2_{block} set. Because $|\alpha|_{N_{cur}} = 0$, no other rules can be used on this form. □

Claim 6. *The second component of* Γ *rewrites any string of the form* ${}^{\triangledown}X_{<}\,{}_{|}X_{0|} \dots {}_{|}X_{j-1|}\,{}_{>}X_{j|} \dots {}_{|}X_{n|}$ *to a string of the form* ${}^{\triangle}XX_1 \dots X_n$*, where* $X_i \in N \cup N_T$*, for all* $i : 0 \leq i \leq n$ *if and only if* ${}_{|}X_{0|} \dots {}_{|}X_{j-1|} = \varepsilon$*; otherwise, blocking symbols are introduced.*

Proof. Proof of Claim 6 is similar to the proof of Claim 5 and it is left to the reader. □

Claim 7. *The second component of* Γ *rewrites any string of the form* ${}^{\triangledown}X_{|}\,{}_{|}X_{0|} \dots {}_{|}X_{j<}\,{}_{|}X_{j+1|} \dots {}_{|}X_{k-1|}\,{}_{>}X_{k|} \dots {}_{|}X_{n|}$ *to a string of the form* ${}^{\triangle}XX_1 \dots X_n$*, where* $X_i \in N \cup N_T$*, for all* $i : 0 \leq i \leq n$ *if and only if* ${}_{|}X_{j+1|} \dots {}_{|}X_{k-1|} = \varepsilon$*; otherwise blocking symbols are introduced.*

Proof. It is similar to the proof of Claim 5 and it is left to the reader. □

Based on the previous claims, it is easy to show that each simulation of a rule of G consists of a single activation of the first component followed by a single activation of the second component of Γ. If the simulated context-sensitive rule is applied in a scattered way, blocking symbols are introduced to the sentential form; otherwise the sentential form is prepared for the simulation of another rule. In the end, all nonblocking symbols are rewritten to terminals thus producing a sentence of the simulated language. Therefore, $L(G) = L(\Gamma)$.

Example 3. *Suppose CSG* $G = (\{A, B, C, D, E\}, \{b, c, d, e\}, P, A)$ *with rules* $P = \{A \to BC, C \to CD, BD \to DB, CD \to ED, B \to b, C \to c, D \to d, E \to e\}$*. Observe that there is no sentential form that could be generated by grammar* G *where the rule* $BD \to DB$ *could be applied.*

Based on the described constructions, equivalent SCGS Γ *can be created as* $\Gamma = (N_{GS}, T, {}^{\triangle}A, P_1, P_2)$*. Now, we try to show, how would* Γ *simulate* G*. Because the amount of rules and symbols created by the transformation algorithm is quite large, we will not list elements of these sets.*

The only rule of G *that has the starting symbol on its left-hand side is* $A \to BC$*. Similarly, only rule applicable on* BC *(we will ignore rules with terminals) is rule* $C \to CD$*. Derivation* $A \Rightarrow^* BCD$ *would be simulated using following sequence of* *derivation steps:*

$$
\begin{aligned}
{}^{\triangle}A &\Rightarrow {}^{\triangledown}B_{|}\,{}_{|}C_{|} && [({}^{\triangle}A) \rightarrow ({}^{\triangledown}B_{|}\,{}_{|}C_{|}) \in P^{1}_{AtoBC}]\\
&\Rightarrow {}^{\triangle}B^{\wedge}_{|}\,{}_{|}C_{|} && [({}^{\triangledown}B_{|}) \rightarrow ({}^{\triangle}B^{\wedge}_{|}) \in P^{2}_{init}]\\
&\Rightarrow {}^{\triangle}B\;C^{\wedge}_{|} && [({}^{\triangle}B^{\wedge}_{|}, {}_{|}C_{|}) \rightarrow ({}^{\triangle}B, C^{\wedge}_{|}) \in P^{2}_{checkf}]\\
&\Rightarrow {}^{\triangle}BC && [({}^{\triangle}B, C^{\wedge}_{|}) \rightarrow ({}^{\triangle}B, C) \in P^{2}_{end}]
\end{aligned}
$$

This way, the first rule is simulated. It is important to note that since Γ *works in* t *mode, rules from set* P_2 *are all applied together. The derivation would continue using the following rules:*

$$
\begin{aligned}
{}^{\triangle}BC &\Rightarrow {}^{\triangledown}B_{|}\,{}_{|}C_{|}\,{}_{|}D_{|} && [({}^{\triangle}B, C) \rightarrow ({}^{\triangledown}B_{|}, {}_{|}C_{|}\,{}_{|}D_{|}) \in P^{1}_{AtoBC}]\\
&\Rightarrow {}^{\triangle}B^{\wedge}_{|}\,{}_{|}C_{|}\,{}_{|}D_{|} && [({}^{\triangledown}B_{|}) \rightarrow ({}^{\triangle}B^{\wedge}_{|}) \in P^{2}_{init}]\\
&\Rightarrow {}^{\triangle}B\;C^{\wedge}_{|}\,{}_{|}D_{|} && [({}^{\triangle}B^{\wedge}_{|}, {}_{|}C_{|}) \rightarrow ({}^{\triangle}B, C^{\wedge}_{|}) \in P^{2}_{checkf}]\\
&\Rightarrow {}^{\triangle}BC\;D^{\wedge}_{|} && [(C^{\wedge}_{|}, {}_{|}D_{|}) \rightarrow (C, D^{\wedge}_{|}) \in P^{2}_{check}]\\
&\Rightarrow {}^{\triangle}BCD && [({}^{\triangle}B, D^{\wedge}_{|}) \rightarrow ({}^{\triangle}B, D) \in P^{2}_{end}]
\end{aligned}
$$

As was mentioned before, rule $BD \rightarrow DB$ *can in fact never be applied by the grammar* G*. Suppose sentential form* ${}^{\triangle}BCD$ *of the* Γ*. Simulation of this rule would lead to the following derivation:*

$$
\begin{aligned}
{}^{\triangle}BCD &\Rightarrow {}^{\triangledown}D_{<}C\,{}_{>}B_{|} && [({}^{\triangle}B, D) \rightarrow ({}^{\triangledown}D_{<}, {}_{>}B_{|}) \in P^{1}_{ABtoCD}]\\
&\Rightarrow {}^{\triangledown}D_{<}\,{}_{|}C_{|}\,{}_{>}B_{|} && [({}^{\triangledown}D_{<}, C) \rightarrow ({}^{\triangledown}D_{<}, {}_{|}C_{|}) \in P^{1}_{phase2}]\\
&\Rightarrow {}^{\triangle}D^{\wedge}_{<}\,{}_{|}C_{|}\,{}_{>}B_{|} && [({}^{\triangledown}D_{<}) \rightarrow ({}^{\triangle}D^{\wedge}_{<}) \in P^{2}_{init}]\\
&\Rightarrow {}^{\triangle}D\,{}_{|}C_{|}\;B^{\wedge}_{|} && [({}^{\triangle}D^{\wedge}_{<}, {}_{>}B_{|}) \rightarrow ({}^{\triangle}D, B^{\wedge}_{|}) \in P^{2}_{checkf}]\\
&\Rightarrow {}^{\triangle}B\,{}_{|}C_{|}D && [({}^{\triangle}B, D^{\wedge}_{|}) \rightarrow ({}^{\triangle}B, D) \in P^{2}_{end}]\\
&\Rightarrow {}^{\triangle}B!D && [(B) \rightarrow (!) \in P^{2}_{block}]
\end{aligned}
$$

Again, each component of Γ *works in* t *mode. This ensures that any symbols skipped during the checking phase, will be replaced by blocking symbols (!) before the second component of* Γ *deactivates.*

On the other hand, rule $CD \rightarrow ED$ *can be applied. The simulation of this rule works as follows:*

$$
\begin{aligned}
{}^{\triangle}BCD &\Rightarrow {}^{\triangledown}B_{|}\,{}_{|}E_{<}\,{}_{>}D_{|} && [({}^{\triangle}B, C, D) \rightarrow ({}^{\triangledown}B_{|}, {}_{|}E_{<}, {}_{>}D_{|}) \in P^{1}_{ABtoCD}]\\
&\Rightarrow {}^{\triangle}B^{\wedge}_{|}\,{}_{|}E_{<}\,{}_{>}D_{|} && [({}^{\triangledown}B_{|}) \rightarrow ({}^{\triangle}B^{\wedge}_{|}) \in P^{2}_{init}]\\
&\Rightarrow {}^{\triangle}B\;E^{\wedge}_{<}\,{}_{>}D_{|} && [({}^{\triangle}B^{\wedge}_{|}, {}_{|}E_{<}) \rightarrow ({}^{\triangle}B, E^{\wedge}_{<}) \in P^{2}_{checkf}]\\
&\Rightarrow {}^{\triangle}BE\;D^{\wedge}_{|} && [(E^{\wedge}_{<}, {}_{>}D_{|}) \rightarrow (E, D^{\wedge}_{|}) \in P^{2}_{check}]\\
&\Rightarrow {}^{\triangle}BED && [({}^{\triangle}B, D^{\wedge}_{|}) \rightarrow ({}^{\triangle}B, D) \in P^{2}_{end}]
\end{aligned}
$$

Theorem 4. $\mathscr{L}(SCGS) = \mathscr{L}(CS)$

Proof. This is implied by Lemmas 1 and 2. □

Theorem 5. *Any context-sensitive language can be generated by a SCGS* Γ *such that* $\mathrm{mcs}(P_i) \leq 1$ *for all components* P_i *of* Γ.

Proof (Basic Idea) 1. *Obviously, only the first subset of* P^1_{ABtoCD} *(see the proof of Lemma 2) has more than two components in its rules. Rules of this subset can be simulated by introduction of some auxiliary rules and symbols. Suppose rule* $p : ({}^{\triangle}X, A, B) \rightarrow ({}^{\triangledown}X_{|}, {}_{|}C_{<,>}D_{|})$ *and sentential form* ${}^{\triangle}XAB$. *This rule can be simulated by using those auxiliary rules in a following way:*

$$ {}^{\triangle}XAB \Rightarrow {}^{p}_{|}X_{|}\,{}_{|}C_{<}B \Rightarrow {}^{\triangledown}X_{|}\,{}_{|}C_{<\,>}D_{|} $$

where always pairs of symbols are rewritten during each derivation step. Where ${}^{p}_{|}X_{|}$ *encodes which rule is being simulated.*

Formally, the set $P^1_{ABtoCD_{mod}}$ would be defined as (appropriate modifications of the alphabet of nonterminals are left to the reader)

$$
\begin{aligned}
P^1_{ABtoCD_{mod}} = &\{({}^{\triangle}X, A) \rightarrow ({}^{p}_{|}X_{|}, {}_{|}C_{<}) : \\
&\quad \forall X \in N_T \cup N, \forall p \in P, p = AB \rightarrow CD\} \\
&\cup \{({}^{p}_{|}X_{|}, B) \rightarrow ({}^{\triangledown}X_{|}, {}_{>}D_{|}) : \\
&\quad \forall X \in N_T \cup N, \forall p \in P, p = AB \rightarrow CD\} \\
&\cup \{({}^{\triangle}A, B) \rightarrow ({}^{\triangledown}C_{<,>}D_{|}) : \\
&\quad \forall p \in P, p = AB \rightarrow CD\}
\end{aligned}
$$

Claim 8. *This change to* $P^1_{ABtoCD_{mod}}$ *only affects the first phase of the first component.*

Proof. From the definition of Γ, only the first component is able to use the rules from the set $P^1_{ABtoCD_{mod}}$. Since the rules from the first and third subset of the $P^1_{ABtoCD_{mod}}$ require the presence of some symbol ${}^{\triangle}X$, these rules can be used only during the first phase of the first component's activation (recall that the second phase requires a presence of some symbol ${}^{\triangledown}X_{|}$ or ${}^{\triangledown}X_{<}$).

The rules of the second subset on the other hand need some symbol ${}^{p}_{|}X_{|}$, however such a symbol can only be introduced by some rule of the first subset and only rules of the second subset are able to rewrite such symbol. □

Claim 9. *The first component of* Γ *rewrites sentential forms of the form* ${}^{\triangle}X\alpha$ *to a string of one of the following forms*

(1) ${}^{\triangledown}Y_{|}\beta$

(2) $^{\triangledown}Y_{<}\gamma$

where $X, Y \in N \cup N_T$, $\alpha \in (N \cup N_T)^*$, $\beta, \gamma \in (N_{CS} \cup N_{CF})^*$ *(such that Claim 1 holds) where exactly one of (a), (b) or (c) given next is true:*

(a) $\beta \in (N_{CF})^*$
(b) $\beta = Y_0 \dots {}_{|}U_{<} \dots {}_{>}V_{|} \dots Y_n$, *where* $Y_i \in N_{CF}, 0 \leq i \leq n$ *and* ${}_{|}U_{<}, {}_{>}V_{|} \in N_{CS}$
(c) $\beta = Y_0 \dots {}_{>}U_{|} \dots {}_{|}V_{<} \dots Y_n$, *where* $Y_i \in N_{CF}, 0 \leq i \leq n$ *and* ${}_{|}U_{<}, {}_{>}V_{|} \in N_{CS}$

and $\gamma = Y_0 \dots {}_{>}V_{|} \dots Y_n$, *where* $Y_i \in N_{CF}, 0 \leq i \leq n$ *and* ${}_{>}V_{|} \in N_{CS}$.

Proof. Since the case (a) above does not deal with context free rewriting, its proof is identical to the proof of Claim 3. What slightly differs from the proof of Claim 3 are the cases (b) and (c).

In addition to the eight possible sentential forms derivable from the $^{\triangle}X\alpha$ described in the proof of Claim 3, the ninth one is introduced:

(1) $^{\triangle}XX_0 \dots X_{i-1}AX_{i+1} \dots X_n$
$\Rightarrow {}^{\triangledown}X_{|}X_0 \dots X_{i-1}\,{}_{|}B_{|}\,{}_{|}C_{|}X_{i+1} \dots X_n$
(2) $^{\triangle}AX_0 \dots X_n$
$\Rightarrow {}^{\triangledown}B_{|}\,{}_{|}C_{|}X_0 \dots X_n$
(3) $^{\triangle}XX_0 \dots X_{i-1}AX_{i+1} \dots X_n$
$\Rightarrow {}^{\triangledown}X_{|}X_0 \dots X_{i-1}\,{}_{|}B_{|}X_{i+1} \dots X_n$
(4) $^{\triangle}AX_0 \dots X_n$
$\Rightarrow {}^{\triangledown}B_{|}X_0 \dots X_n$
(5) $^{\triangle}XX_0 \dots X_{i-1}AX_{i+1} \dots X_n$
$\Rightarrow {}^{\triangledown}X_{|}X_0 \dots X_{i-1}\,{}_{|}a'_{|}X_{i+1} \dots X_n$
(6) $^{\triangle}AX_0 \dots X_n$
$\Rightarrow {}^{\triangledown}a'_{|}X_0 \dots X_n$
(7) $^{\triangle}XX_0 \dots X_{i-1}A\delta BX_{i+1} \dots X_n$
$\Rightarrow^2 {}^{\triangledown}X_{|}X_0 \dots X_{i-1}\,{}_{|}C_{<}\delta\,{}_{>}D_{|}X_{i+1} \dots X_n$
(8) $^{\triangle}A\delta BX_0 \dots X_n$
$\Rightarrow {}^{\triangledown}C_{<}\delta\,{}_{>}D_{|}X_0 \dots X_n$
(9) $^{\triangle}XX_0 \dots X_{i-1}B\delta AX_{i+1} \dots X_n$
$\Rightarrow^2 {}^{\triangledown}X_{|}X_0 \dots X_{i-1}\,{}_{>}D_{|}\delta\,{}_{|}C_{<}X_{i+1} \dots X_n$

where $\delta \in (N \cup N_T)^*$.

Of those sentential forms, only the options 7 and 9 need to be further analyzed, the remaining ones work in a same manner as they did in Claim 3.

Suppose sentential form $^{\triangle}X\alpha$, where $\alpha = X_0 \dots X_n$ and the set of rules $P^1_{ABtoCD_{mod}}$ defined as above (to be more specific, only its first two subsets). Using some rule of the first subset, the derivation of the following form may occur:

$$\begin{aligned} &{}^{\triangle}X X_0 \ldots X_{i-1}\alpha_1 A \alpha_2 X_{i+1} \ldots X_n \\ &\Rightarrow {}^{p}_{|}X_{|} X_0 \ldots X_{i-1}\alpha_1\, {}_{|}C_{<}\alpha_2 X_{i+1} \ldots X_n \end{aligned}$$

where either $\alpha_1 = \delta B$ and $\alpha_2 = \varepsilon$ or $\alpha_1 = \varepsilon$ and $\alpha_2 = B\delta$. Since the first symbol of this sentential form is marked as ${}^{p}_{|}X_{|}$ and no other marks are present in the sentential form, only some rule from the second subset may be used and more specifically, only the rule which was created from the rule p. Since the first and second second symbol of the context sensitive rule is rewritten separately, one of the following two possibilities may occur:

$$ {}^{\triangledown}X_{|} X_0 \ldots X_{i-1}\, {}_{|}C_{<}\delta\, {}_{>}D_{|} X_{i+1} \ldots X_n \tag{1}$$

$$ {}^{\triangledown}X_{|} X_0 \ldots X_{i-1}\, {}_{>}D_{|}\delta\, {}_{|}C_{<} X_{i+1} \ldots X_n \tag{2}$$

Observe that these two options correspond to the sentential forms 7 and 9, respectively.

The rest of the proof is the same as in the case of the proof of Claim 3, so it is left to the reader. □

Claim 10. *The sentential form newly introduced in the previous claim (the case (c)) will not lead to generation of a sentence.*

Proof. The careful examination of the set P_2 shows that the symbols $C^{\wedge}_{<}$ and ${}_{>}D_{|}$ appear only in that order on the left-hand side of the checking rules. If the symbols appear in the opposite order (i.e. symbol expecting left context sensitivity appearing first), there are no rules that would be able to rewrite them. These symbols would thus be skipped during the checking phase, and blocking symbols would be introduced instead. □

Proof. Based on the proof of Theorem 4 with the modification introduced by Claims 8, 9 and 10, Theorem 5 holds. □

4. Conclusion

The modified version of $\mathscr{L}(PSCG) = \mathscr{L}(CS)$ problem was discussed in this paper. This modification deals with combination of CD grammar systems with propagating scattered context components and compares their generative power with context-sensitive grammars. The algorithm that constructs grammar system that simulates given context-sensitive grammar has been described. Based on this algorithm, it is shown that those two models have the same generative power. Furthermore it is shown that this property holds even for the most simple variant of these grammar systems—that is, those using only two components, where each scattered context rule is of degree of at most two.

Acknowledgment

This work was supported by The Ministry of Education, Youth and Sports of the Czech Republic from the National Programme of Sustainability (NPU II); project IT4Innovations excellence in science - LQ1602; the TAČR grant TE01020415; and the BUT grant FIT-S-17-3964.

References

[1] E. Csuhaj-Varjú and J. Dassow, On cooperating/distributed grammar systems, *Journal of Information Processing and Cybernetics* **26**(1-2) (1990) 49–63.
[2] E. Csuhaj-Varju, J. Kelemen, G. Paun and J. Dassow, *Grammar Systems: A Grammatical Approach to Distribution and Cooperation* (Gordon and Breach Science Publishers, 1994).
[3] J. Dassow and G. Păun, *Regulated rewriting in formal language theory* (Springer, 1989).
[4] J. Gonczarowski and M. K. Warmuth, Scattered versus context-sensitive rewriting, *Acta Informatica* **27** (Nov 1989) 81–95.
[5] S. Greibach and J. Hopcroft, Scattered context grammars, *Journal of Computer and System Sciences* **3**(3) (1969) 233–247.
[6] S.-Y. Kuroda, Classes of languages and linear-bounded automata, *Information and Control* **7**(2) (1964) 207–223.
[7] A. Meduna, *Automata and Languages: Theory and Applications* (Springer, 2000).
[8] A. Meduna and J. Techet, *Scattered Context Grammars and their Applications* (WIT Press, 2010).
[9] G. Rozenberg and A. Salomaa, *Handbook of Formal Languages, Vol. 1: Word, Language, Grammar* (Springer, 1997).
[10] G. Rozenberg and A. Salomaa, *Handbook of Formal Languages, Vol. 2: Linear Modeling: Background and Application* (Springer, 1997).