Rule-Homogeneous CD Grammar Systems

Radim Kocman Zbyněk Křivka Alexander Meduna
Centre of Excellence IT4Innovations
Faculty of Information Technology
Brno University of Technology
Božetěchova 2, Brno 612 66, Czech Republic
{ikocman,krivka,meduna}@fit.vutbr.cz

A homogeneous rule has its left-hand side formed by a string of identical symbols. Consider two-component CD grammar systems that work under the $*$ mode or the t mode. This paper demonstrates that the family of recursively enumerable languages is characterized by these systems in which one component is context-free and the other has two evenly homogeneous rules—$11 \to 00$ and $0000 \to 2222$, where 0, 1, and 2 are nonterminals. Furthermore, this characterization also holds in terms of two-component CD grammar systems in which one component is context-free and the other has two homogeneous rules—$11 \to 00$ and $0000 \to \varepsilon$. Several properties concerning these systems are formulated and studied.

1 Introduction

The present paper, which assumes a familiarity with formal language theory (see [5, 10]), concerns grammar systems (see [2]). It concentrates its attention on two-component CD grammar systems working under the $*$ and t modes. Under the former mode, these systems obviously generate the family of context-free languages. More surprisingly, under the latter mode, they are also as powerful as ordinary context-free grammars, too. These results give rise to an idea of changing one of the context-free components with a simple non-context-free component so this change results into an increase of the generative power. An investigation of this idea represents the principal subject of the present study.

Before sketching this study and its achievement, we recall that a grammatical rule of the form $x \to y$, where x and y are strings, is homogeneous if x is formed by a string of identical symbols (see [6]). A homogeneous rule $x \to y$ is evenly homogeneous if y is also formed by a string of identical symbols and $|x| = |y|$. In a CD grammar system, a component is homogeneous if all its rules are homogeneous, and it is evenly homogeneous if all its rules are evenly homogeneous. A CD grammar system is rule-homogeneous if all its components are homogeneous. Observe that any CD grammar system with context-free components is rule-homogeneous. As obvious, if a CD grammar system contains only evenly homogeneous rules, its language is a subset of the terminal alphabet in the system.

To give an insight into the present study, take any grammar G in Kuroda normal form—that is, a grammar in which every rule is of the form $AB \to CD$, $A \to BC$, $A \to a$, or $A \to \varepsilon$, where A, B, C, D are nonterminals, a is a terminal, and ε denotes the empty string (see Section 8.3.3. in [5]). This paper demonstrates two transformations that turn G to a two-component rule-homogeneous CD grammar system with a context-free component H and a homogeneous non-context-free component I. One transformation produces $I = \{11 \to 00,\ 0000 \to 2222\}$, so I is evenly homogeneous. The other transformation produces $I = \{11 \to 00,\ 0000 \to \varepsilon\}$, which is thus homogeneous. The paper proves that working under the $*$ and t modes, both systems resulting from these transformations are equivalent to G. Thus, more generally speaking, CD grammar systems of these two forms are computationally

Submitted to:
AFL 2017

complete—that is, they characterize the family of recursively enumerable languages—because so are the Kuroda normal form grammars.

Apart from the computational completeness, it is worth mentioning two other properties, (i) and (ii), concerning the t mode and the $*$ mode, respectively.

(i) Consider the system with $I = \{11 \to 00,\ 0000 \to \varepsilon\}$. The paper demonstrates that working under the t mode, during every generation of a sentence, it changes its components no more than once. Furthermore, if the system simulates at least one non-context-free rule, it changes its components precisely once.

(ii) From a general and intuitive viewpoint, taking a closer look at language-generating rewriting systems, we intuitively see that some of them generate the language in a more similar way than others. More precisely, consider models X and Y. If there is a constant k such that for every derivation of the form

$$x_0 \Rightarrow x_1 \Rightarrow \ldots \Rightarrow x_n$$

in X, where x_0 is its start symbol, there is a derivation of the form

$$x_0 \Rightarrow^{k_1} x1 \Rightarrow^{k_2} \ldots \Rightarrow^{k_n} x_n$$

in Y, where $k_i \leq k$ for each $1 \leq i \leq n$, we tend to say that Y closely simulates X. In this sense, the paper demonstrates that under the $*$ mode, the system with $I = \{11 \to 00,\ 0000 \to 2222\}$ closely simulates G, and so does the system with $I = \{11 \to 00,\ 0000 \to \varepsilon\}$ under this mode.

The rest of the paper is organized as follows. Section 2 recalls all the terminology needed in this paper and introduces the notion of rule-homogeneous CD grammar systems. Section 3 then presents all fundamental results achieved in this study as sketched above.

2 Preliminaries and Definitions

This paper assumes that the reader is familiar with the theory of automata and formal languages (see [5, 10]). This section recalls only the crucial notions used in this paper.

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. Members of V^* are called *strings*. Set $V^+ = V^* - \{\varepsilon\}$; algebraically, V^+ is thus the free semigroup generated by V under the operation of concatenation. For $x \in V^*$, $|x|$ denotes the length of x, $\mathrm{rev}(x)$ denotes the reversal of x, and $\mathrm{alph}(x)$ denotes the set of all symbols occurring in x; for instance, $\mathrm{alph}(0010) = \{0,1\}$. Let **CF**, **CS**, and **RE** denote the families of context-free, context-sensitive, and recursively enumerable languages, respectively.

A *phrase-structure grammar* or, more simply, a *grammar* is a quadruple, $G = (N,T,P,S)$, whose components are defined as follows. N and T are alphabets such that $N \cap T = \emptyset$. Symbols in N are referred to as *nonterminals*, while symbols in T are referred to as *terminals*. $S \in N$ is the start symbol of G. P is a finite set of productions (rules) such that every $p \in P$ has the form $x \to y$, where $x, y \in (N \cup T)^*$ and $\mathrm{alph}(x) \cap N \neq \emptyset$; the left-hand side x and the right-hand side y of p are denoted by $\mathrm{lhs}(p)$ and $\mathrm{rhs}(p)$, respectively. The rule $p \in P$ is considered context-free if $|\mathrm{lhs}(p)| = 1$; otherwise, it is a non-context-free rule. If $x \to y \in P$ and $u, w \in (N \cup T)^*$, then $uxw \Rightarrow uyw$. In the standard manner, extend $\Rightarrow$ to $\Rightarrow^n$, where $n \geq 0$; then, based on $\Rightarrow^n$, define $\Rightarrow^+$ and $\Rightarrow^*$. The *language generated by* G, $L(G)$, is defined as $L(G) = \{w \in T^* \mid S \Rightarrow^* w\}$.

Let $G = (N, T, P, S)$ be a grammar. G is in *Kuroda normal form* (see Section 8.3.3. in [5]) if every rule $p \in P$ has one of these three forms: (1) $AB \to CD$, where $A, B, C, D \in N$, (2) $A \to BC$, where $A, B, C \in N$, or (3) $A \to a$, where $A \in N$ and $a \in (T \cup \{\varepsilon\})$. If $x \to y \in P$ and $x \in \{A\}^+$ for some $A \in N$, then $x \to y$ is a *homogeneous rule* (see [6]). Furthermore, if also $y \in \{B\}^+$ for some $B \in (N \cup T)$ and $|x| = |y|$, then $x \to y$ is an *evenly homogeneous rule*. G represents a *homogeneous grammar* if every $p \in P$ is homogeneous. Lastly, set $\text{ContextFree}(P) = \{p \in P \mid |\text{lhs}(p)| = 1\}$ and $\text{NonContextFree}(P) = \{p \in P \mid |\text{lhs}(p)| \geq 2\}$.

A *phrase-structure cooperating distributed grammar system* (a *phrase-structure CD grammar system* for short) is a construct $\Gamma = (N, T, P_1, P_2, \ldots, P_n, S)$, $n \geq 1$, where N is the alphabet of nonterminals, T is the alphabet of terminals, $N \cap T = \emptyset$, $S \in N$ is the start symbol, and for $1 \leq i \leq n$, each component P_i is a finite set of phrase-structure rules. (For the original context-free definition see [2].) For $u, v \in V^*$, $V = N \cup T$, and $1 \leq k \leq n$, let $u \Rightarrow_{P_k} v$ denote a derivation step performed by the application of a rule from P_k. As usual, extend the relation $\Rightarrow_{P_k}$ to $\Rightarrow^m_{P_k}$ (the m-step derivation), $m \geq 0$, $\Rightarrow^+_{P_k}$, and $\Rightarrow^*_{P_k}$. In addition, we define the relation $u \Rightarrow^t_{P_k} v$ so that $u \Rightarrow^*_{P_k} v$ and there is no $w \in V^*$ such that $v \Rightarrow_{P_k} w$. The language generated by Γ working in the f mode, $f \in \{*, t\}$, denoted by $L_f(\Gamma)$, is defined as $L_f(\Gamma) = \{w \in T^* \mid S \Rightarrow^f_{P_{k_1}} w_1 \Rightarrow^f_{P_{k_2}} \ldots \Rightarrow^f_{P_{k_l}} w_l = w,\ l \geq 1,\ 1 \leq k_i \leq n,\ 1 \leq i \leq l\}$. Γ is referred to as *rule-homogeneous*, *evenly rule-homogeneous*, or *context-free* (instead of phrase-structure) if all its rules are homogeneous, evenly homogeneous, or context-free, respectively.

Language families generated by context-free CD grammar systems with n components working in the f mode and allowing ε-rules are denoted by $CD^{\varepsilon}_n(f)$. When the number of components is not limited, we replace n by ∞. The following results are well-known (see Theorem 3.1 in [8]):

(i) $CD^{\varepsilon}_{\infty}(*) = \mathbf{CF}$,

(ii) $\mathbf{CF} = CD^{\varepsilon}_1(t) = CD^{\varepsilon}_2(t) \subset CD^{\varepsilon}_3(t) = CD^{\varepsilon}_{\infty}(t) = \mathbf{ET0L}$,

where **ET0L** denotes the family of languages generated by extended tabled interactionless Lindenmayer systems (see [7]).

The definition of phrase-structure CD grammar systems can be easily modified so that the components are sets of rules of any arbitrary type. Recall that for CD grammar systems having regular, linear, context-sensitive, or phrase-structure components, the generative power does not change (see [2, 8]), i.e., they generate the families of regular, linear, context-sensitive, or recursively enumerable languages, respectively. Nonetheless, different results have been obtained by studying some other non-classical components—e.g., permitting, left-forbidding, and random context components (see [1, 3, 4])—where the number of components affects the resulting generative power.

It is clear that if we require a significant increase in the generative power, we need components that use a stronger mechanism than basic context-free rules. In general, components with homogeneous rules have the similar effect as phrase-structure components—a single homogeneous component can define **RE** by itself (see [6]). The same, however, does not hold for components with evenly homogeneous rules, which can clearly generate only single symbol results on their own. Therefore, one may wonder, if we combine several components of different types, how simple the additional non-context-free part can be so it still significantly increases the generative power of context-free CD grammar systems.

The rest of the paper studies two-component rule-homogeneous CD grammar systems where the first component is context-free and the second component contains either evenly homogeneous or homogeneous rules. Furthermore, we limit the non-context-free component so it contains only two rules.

3 Results

First, let us start with the most straightforward variant of a two-component rule-homogeneous CD grammar system that works in the $*$ mode and has the second component homogeneous. The following proof will also serve as a framework for the later proofs since quite a few parts of the reasoning are shared throughout the variants.

Theorem 1. *Let $G = (N,T,P,S)$ be a grammar. Then, there exists a two-component rule-homogeneous CD grammar system, $\Gamma = (N',T,H,I,S)$, such that $I = \{11 \to 00,\ 0000 \to \varepsilon\}$ and $L_*(\Gamma) = L(G)$.*

Proof.
Construction. Let $G = (N,T,P,S)$ be a grammar. Without any loss of generality, assume that G satisfies the Kuroda normal form and $(N \cup T) \cap \{0,1\} = \emptyset$. For some $m \geq 3$, define an injection, g, from NonContextFree(P) to $(\{01\}^+\{00\}\{01\}^+ \cap \{01,00\}^m)$.

From G, we construct the two-component rule-homogeneous CD grammar system, $\Gamma = (N',T,H,I,S)$, where $N' = N \cup \{0,1\}$, $I = \{11 \to 00,\ 0000 \to \varepsilon\}$, and H is defined as follows:

(I) For every $AB \to CD \in P$ where $A,B,C,D \in N$,
add $A \to Cg(AB \to CD)$ and $B \to \mathrm{rev}(g(AB \to CD))D$ to H.

(II) For every $A \to x \in P$ where $A \in N$ and $x \in (\{\varepsilon\} \cup T \cup N^2)$, add $A \to x$ to H.

The construction of Γ is completed.

Basic idea.

(a) The rules of (I) and I simulate the derivation steps made by NonContextFree(P) in G. That is, $xABy \Rightarrow xCDy$ according to $AB \to CD \in P$, where $x,y \in (N \cup T)^*$, in G is simulated in Γ as

$$\begin{aligned} xABy &\Rightarrow_H xCg(AB \to CD)By \\ &\Rightarrow_H xCg(AB \to CD)\,\mathrm{rev}(g(AB \to CD))Dy \\ &\Rightarrow_I^{2m-1} xCDy. \end{aligned}$$

Γ makes the $(2m-1)$-step derivation $xCg(AB \to CD)\,\mathrm{rev}(g(AB \to CD))Dy \Rightarrow_I^{2m-1} xCDy$ by using only rules from $I = \{11 \to 00,\ 0000 \to \varepsilon\}$. During this $(2m-1)$-step derivation, the string between C and D always contains exactly one occurrence of consecutive identical symbols that can be rewritten, so Γ actually verifies that the simulation of $xABy \Rightarrow xCDy$ is made properly.

(b) The rules of (II) simulate the use of ContextFree(P) in G.

The reader may notice that the simulation of non-context-free rules resembles similar techniques used in phrase-structure grammars (see [9, 6]). However, this is traditionally done by using several types of matching parentheses (see [9]), which is not a suitable form for homogeneous rules, or it requires a significant non-local change in the generation flow of the grammar (see [6]).

Formal proof (sketch). We prove $L_*(\Gamma) = L(G)$. First, we prove that the verification process of the simulation is valid and cannot be disturbed.

Claim 2. The verification process of simulated NonContextFree(P) in Γ is valid and cannot be disturbed by other rules.

Proof. Consider any $AB \to CD \in P$ and any derivation step $xABy \Rightarrow xCDy$ in G, where $A,B,C,D \in N$ and $x,y \in (N \cup T)^*$. For some $m \geq 3$, this is simulated in Γ with rules $A \to C01(01)^k00(01)^l01$ and $B \to 10(10)^l00(10)^k10D$, where $k,l \geq 0$ and $k+l+3=m$. The result of their correct application can generally be in the form $uC01(01)^k00(01)^l01w10(10)^l00(10)^k10Dv$, where $u,v,w \in (N' \cup T)^*$. Observe that parts u, v, w, C, and D can potentially contain and generate some additional nonterminals 0 and 1. Nonetheless, these nonterminals can be generated only from the previous two rules or from the other simulated rules in the forms $01(01)^p00(01)^q01$ and $10(10)^q00(10)^p10$, where $p,q \geq 0$, $p+q+3=m$, and $p \neq k$. We show that, in any situation, the verification process holds.

(1) Consider the simplest case where all parts u, v, w, C, and D ended as ε. We begin the process with $01(01)^k00(01)^l0110(10)^l00(10)^k10$. First, we use $l+1$ times rules $11 \to 00$ and $0000 \to \varepsilon$, respectively, and get $01(01)^k0000(10)^k10$. Next, we use the rule $0000 \to \varepsilon$, which in fact verifies the match of both parts. And lastly, we use $k+1$ times both rules again to erase the rest. Observe that these nonterminals cannot be processed in any other way, and that this process can start only if the verification parts from both rules meet each other.

(2) Now consider cases where the simulation is done incorrectly. Without a loss of generality, assume only the sequences of nonterminals 0 and 1. Other symbols can only further block the process.

(2.1) If there is only one part $01(01)^k00(01)^l01$ or $10(10)^l00(10)^k10$ alone, it cannot be erased.

(2.2) If two parts $10(10)^l00(10)^k10$ and $01(01)^k00(01)^l01$ meet in the wrong order, the verification process cannot start since there are no possible derivation steps. The same holds if two parts from different rules meet in the wrong order.

(2.3) If two different parts $01(01)^k00(01)^l01$ and $10(10)^q00(10)^p10$ meet in the proper order, the process gets stuck. Assume that $l > q$. We begin with $01(01)^k00(01)^l0110(10)^q00(10)^p10$ and use $q+1$ times rules $11 \to 00$ and $0000 \to \varepsilon$, respectively. This ends with the sequence $01(01)^k00(01)^{l-q}00(10)^p10$ which cannot be processed any further. Note that for $l < q$ the result is analogical. Observe that if the verification process gets stuck in this way, the resulting sequence begins and ends with 0. Therefore, in the same way as in (2.2), it cannot interact with other verification sequences anymore.

(2.4) Any other case is a combination of (2.1), (2.2), and (2.3).

(3) Finally, consider the full case of the form $uC01(01)^k00(01)^l01w10(10)^l00(10)^k10Dv$ where parts u, v, w, C, and D can contain and generate additional symbols. Observe that parts u, v, C, and D cannot affect the process, since no result of (1) and (2) or another verification part can interact with a sequence that begins/ends with 0. Lastly, part w has to always end as ε; otherwise, either the simulation is done incorrectly, and it is in fact some case of (2); or both parts get correctly matched with some different parts in the end, and the same is also possible in G (e.g., if w generates BA, it can simulate some $xABABy \Rightarrow^* xCDCDy$ in G).

Thus, Claim 2 holds. ■

Next, we prove $L(G) \subseteq L_*(\Gamma)$; more precisely, by induction on $i \geq 0$, we demonstrate Claim 3.

Claim 3. For every $w \in (N \cup T)^*$ and $i \geq 0$, $S \Rightarrow^i w$ in G implies $S \Rightarrow^*_{k_1} w_1 \Rightarrow^*_{k_2} \ldots \Rightarrow^*_{k_l} w_l = w$, $l \geq 1$, $k_j \in \{H,I\}$, $1 \leq j \leq l$, in Γ.

Proof. This proof by induction on $i \geq 0$ is very simple. Therefore we omit its basis and only sketch the rest. Assume that the implication of Claim 3 holds for every $i \leq o$, where o is a non-negative integer. Consider any derivation of the form $S \Rightarrow^{o+1} \beta$, where $\beta \in (N \cup T)^*$. Express $S \Rightarrow^{o+1} \beta$ as $S \Rightarrow^o \alpha \Rightarrow \beta$,

where $\alpha \in (N \cup T)^*$. By the induction hypothesis, $S \Rightarrow_{k_1}^* w_1 \Rightarrow_{k_2}^* \ldots \Rightarrow_{k_l}^* w_l = \alpha$, $l \geq 1$, $k_j \in \{H, I\}$, $1 \leq j \leq l$, in Γ. There are the following two possibilities how G can make $\alpha \Rightarrow \beta$:

(1) Let $AB \to CD \in P$, $\alpha = xABy$, $\beta = xCDy$, $x, y \in (N \cup T)^*$, $A, B, C, D \in N$. From (a) and Claim 2,

$$\begin{aligned} xABy &\Rightarrow_H xCg(AB \to CD)By \\ &\Rightarrow_H xCg(AB \to CD)\,\mathrm{rev}(g(AB \to CD))Dy \\ &\Rightarrow_I^{2m-1} xCDy \end{aligned}$$

in Γ. Consequently, $S \Rightarrow_{k_1}^* w_1 \Rightarrow_{k_2}^* \ldots \Rightarrow_{k_l}^* w_l = xCDy = \beta$, $l \geq 1$, $k_j \in \{H, I\}$, $1 \leq j \leq l$, in Γ.

(2) Let $A \to z \in P$, $\alpha = xAy$, $\beta = xzy$, $x, y \in (N \cup T)^*$, $A \in N$, $z \in (\{\varepsilon\} \cup T \cup N^2)$.
This case is left to the reader.

The induction step is completed, so Claim 3 holds. ■

Lastly, we prove $L_*(\Gamma) \subseteq L(G)$. Based on Claim 2, it is rather easy to demonstrate (a rigorous version of this demonstration is left to the reader) that Γ can generate every $y \in L_*(\Gamma)$ as

$$\begin{aligned} S = v_{0_3} &\Rightarrow_H^* v_{1_0} \Rightarrow_H v_{1_1} \Rightarrow_H v_{1_2} \Rightarrow_I^{2m-1} v_{1_3} \\ &\Rightarrow_H^* v_{2_0} \Rightarrow_H v_{2_1} \Rightarrow_H v_{2_2} \Rightarrow_I^{2m-1} v_{2_3} \\ &\vdots \\ &\Rightarrow_H^* v_{k_0} \Rightarrow_H v_{k_1} \Rightarrow_H v_{k_2} \Rightarrow_I^{2m-1} v_{k_3} \Rightarrow_H^* v_{(k+1)_0} = y \end{aligned}$$

where for $i = 0, 1, \ldots, k$ in $v_{i_3} \Rightarrow_H^* v_{(i+1)_0}$ every sentential form is over $(N \cup T)^*$; and for $j = 1, \ldots, k$ the derivation $v_{j_0} \Rightarrow_H v_{j_1} \Rightarrow_H v_{j_2} \Rightarrow_I^{2m-1} v_{j_3}$ has the following form:

$v_{j_0} = u_j A_j B_j w_j$,
$v_{j_1} = u_j C_j g(A_j B_j \to C_j D_j) B_j w_j$,
$v_{j_2} = u_j C_j g(A_j B_j \to C_j D_j)\,\mathrm{rev}(g(A_j B_j \to C_j D_j)) D_j w_j$,
$v_{j_2} \Rightarrow_I^{2m-1} v_{j_3}$ is made by using $11 \to 00$ and $0000 \to \varepsilon$,
$v_{j_3} = u_j C_j D_j w_j$
for some $A_j B_j \to C_j D_j \in P$, $u_j, w_j \in (N \cup T)^*$.

From the derivation of the above form in Γ and from (I) and (II), we see that $v_{0_3} \Rightarrow^* v_{(k+1)_0}$ in G. Therefore, $y \in L_*(\Gamma)$ implies $y \in L(G)$. Thus, $L_*(\Gamma) \subseteq L(G)$.

As $L(G) \subseteq L_*(\Gamma)$ and $L_*(\Gamma) \subseteq L(G)$, $L_*(\Gamma) = L(G)$. Thus, Theorem 1 holds. □

Corollary 4. *The resulting two-component rule-homogeneous CD grammar system Γ from the proof of Theorem 1 closely simulates the original grammar G in Kuroda normal form.*

Proof. For any resulting Γ, we can find a bounded constant k such that for every possible derivation $u \Rightarrow v$ in G there is a k_i-step derivation in Γ that gives the same result and $k_i \leq k$. Furthermore, for a given Γ, we can easily determine the minimal possible k.

Consider the proof of Claim 3 and the mentioned possibilities how G can make $\alpha \Rightarrow \beta$. Clearly, any context-free rule can be simulated in one derivation step. The remaining non-context-free rules require two initial derivation steps and the following verification process. The length of the verification is dependent on the selected size m for the verification code, and it takes $2m-1$ steps to complete. The minimal possible k for a given Γ is therefore $2m+1$. □

Next, we consider a CD grammar system with the same structure but working in the t mode.

Theorem 5. *Let $G = (N,T,P,S)$ be a grammar. Then, there exists a two-component rule-homogeneous CD grammar system, $\Gamma = (N',T,H,I,S)$, such that $I = \{11 \to 00,\ 0000 \to \varepsilon\}$ and $L_t(\Gamma) = L(G)$.*

Proof.
Construction. The process of construction remains identical to Theorem 1. For a grammar $G = (N,T,P,S)$, some $m \geq 3$, and injection g, we construct the two-component rule-homogeneous CD grammar system, $\Gamma = (N',T,H,I,S)$, where $N' = N \cup \{0,1\}$, and H and I contain the rules as described previously.

Basic idea.
Recall that, during the generation of a sentence, a CD grammar system working in the t mode switches its components only if the process is not finished and there are no possible derivations with the previous component. Consider the general behavior of Γ. It starts the generation with S. For the first derivation, applicable rules can be found only in H, so this component has to be used. However, H also contains all rules simulating the original rules of G. Consequently, the first derivation in the t mode has to simulate all rules in G without completing the verification process for non-context-free rules. Nonetheless, we prove that the verification process can be done successfully afterwards for all simulated rules at once.

Formal proof (sketch).
We prove $L_t(\Gamma) = L(G)$. First, let us prove the statement introduced above. For convenience, consider the homomorphism $\varphi : (N' \cup T)^* \to (N \cup T)^*$ where $\varphi(a) = a$ and $\varphi(b) = \varepsilon$, for all $a \in (N \cup T)$ and $b \in \{0,1\}$.

Claim 6. For every $u \in (N \cup T)^*$ and $i \geq 0$, $S \Rightarrow^i u$ in G implies $S \Rightarrow_H^* w \Rightarrow_I^t u$ in Γ, where $w \in (N' \cup T)^*$ and $\varphi(w) = u$. Furthermore, we consider w to be generally in the form $w = p_1 q_1 \dots p_n q_n$, where $n \geq 1$, $p_j \in (N \cup T)^*$, $q_j \in \{0,1\}^*$, $1 \leq j \leq n$, and every q_j represents a string that can be successfully verified and erased by the verification process.

Proof. Basis: Let $i = 0$. Then, $u = S$. Clearly, $S \Rightarrow_H^0 S \Rightarrow_I^t S$, and the required form also holds.
Induction hypothesis: Assume that Claim 6 holds for every $i = 0, \dots, o$, where o is a non-negative integer.

Induction step: Consider any derivation of the form $S \Rightarrow^{o+1} \beta$, where $\beta \in (N \cup T)^*$. Express $S \Rightarrow^{o+1} \beta$ as $S \Rightarrow^o \alpha \Rightarrow \beta$, where $\alpha \in (N \cup T)^*$. By the induction hypothesis, $S \Rightarrow_H^* w \Rightarrow_I^t \alpha$, where $\varphi(w) = \alpha$, in Γ. There are the following two possibilities how G can make $\alpha \Rightarrow \beta$:

(1) Let $AB \to CD \in P$, $\alpha = xABy$, $\beta = xCDy$, $x,y \in (N \cup T)^*$, $A,B,C,D \in N$. Consider w in the required form. Let $w = p_1 q_1 \dots p_k A q_k B p_{k+1} q_{k+1} \dots p_n q_n$, where $n \geq 1$, $1 \leq k \leq n$, $p_j \in (N \cup T)^*$, $q_j \in \{0,1\}^*$, $1 \leq j \leq n$, and also $p_1 \dots p_k = x$ and $p_{k+1} \dots p_n = y$. Then

$$\begin{aligned} & w = p_1 q_1 \dots p_k A q_k B p_{k+1} q_{k+1} \dots p_n q_n \\ & \Rightarrow_H p_1 q_1 \dots p_k C g(AB \to CD) q_k B p_{k+1} q_{k+1} \dots p_n q_n \\ & \Rightarrow_H p_1 q_1 \dots p_k C g(AB \to CD) q_k \operatorname{rev}(g(AB \to CD)) D p_{k+1} q_{k+1} \dots p_n q_n = w' \end{aligned}$$

in Γ, and there are two possible situations regarding these steps:

(a) If $q_k = \varepsilon$, the steps add a new sequence of verification symbols. However, by Claim 2, such a sequence can be successfully verified and cleared on its own, so the required form holds. Consequently, $S \Rightarrow_H^* w' \Rightarrow_I^t \beta$ in Γ.

(b) If $q_k \neq \varepsilon$, the steps prolong some existing sequence of verification symbols. However, by Claim 2, observe that this creates a properly nested structure of verification codes that can also be completely verified and erased on its own, so the required form holds. Consequently, $S \Rightarrow_H^* w' \Rightarrow_I^t \beta$ in Γ.

(2) Let $A \to z \in P$, $\alpha = xAy$, $\beta = xzy$, $x, y \in (N \cup T)^*$, $A \in N$, $z \in (\{\varepsilon\} \cup T \cup N^2)$.
This case is left to the reader.

The induction step is completed, so Claim 6 holds. ■

Consider $S \Rightarrow^* y$, where $y \in T^*$, in G. By Claim 6, this implies $S \Rightarrow_H^* w \Rightarrow_I^t y$, where $w \in (T \cup \{0,1\})^*$, in Γ. It is obvious that, in such a case, $\Rightarrow_H^*$ behaves exactly the same as $\Rightarrow_H^t$. Thus, $L(G) \subseteq L_t(\Gamma)$. Nonetheless, it is clear that Γ working in the t mode can no longer closely simulate G.

The proof for $L_t(\Gamma) \subseteq L(G)$ is just a variation of the proof for $L_*(\Gamma) \subseteq L(G)$ from Theorem 1 with the modified derivation order and required string forms from Claim 6; thus, it is left to the reader.

As $L(G) \subseteq L_t(\Gamma)$ and $L_t(\Gamma) \subseteq L(G)$, $L_t(\Gamma) = L(G)$. Thus, Theorem 5 holds. □

Corollary 7. *The resulting two-component rule-homogeneous CD grammar system Γ from the proof of Theorem 5 changes its components, during every generation of a sentence, no more than once.*

Proof. This proof directly follows the basic idea of Theorem 5 and Claim 6. Γ starts the process with symbol S and component H, since it is the only component that can generate something from S. If the first derivation does not use any simulated non-context-free rules, Γ never switches components, because the result of such a derivation is already a final sentence. If the verification process is required, then, at the end, Γ switches to component I that finishes the generation. Since I cannot introduce any new nonterminals of the original grammar, Γ is not able to switch again. □

For the remaining results, we change the second component of the two-component rule-homogeneous CD grammar system so it is evenly homogeneous. As previously, the first theorem describes a variant that works in the $*$ mode.

Theorem 8. *Let $G = (N, T, P, S)$ be a grammar. Then, there exists a two-component rule-homogeneous CD grammar system, $\Gamma = (N', T, H, I, S)$, such that $I = \{11 \to 00,\ 0000 \to 2222\}$ and $L_*(\Gamma) = L(G)$.*

Proof.
Construction. Let $G = (N, T, P, S)$ be a grammar. Without any loss of generality, assume that G satisfies the Kuroda normal form and $(N \cup T) \cap \{0,1,2\} = \emptyset$. For some $m \geq 3$, define an injection, g, from NonContextFree(P) to $(\{01\}^+\{00\}\{01\}^+ \cap \{01,00\}^m)$.

From G, we construct the two-component rule-homogeneous CD grammar system, $\Gamma = (N', T, H, I, S)$, where $N' = N \cup \{0,1,2\}$, $I = \{11 \to 00,\ 0000 \to 2222\}$, and H is defined as follows:

(I) For every $AB \to CD \in P$ where $A, B, C, D \in N$,
add $A \to Cg(AB \to CD)$ and $B \to \text{rev}(g(AB \to CD))D$ to H.

(II) For every $A \to x \in P$ where $A \in N$ and $x \in (\{\varepsilon\} \cup T \cup N^2)$, add $A \to x$ to H.

(III) Add $2 \to \varepsilon$ to H.

The construction of Γ is completed.

Note that this resembles the construction from Theorem 1. We only add one additional nonterminal and a rule that can erase it. Also the basic idea for the simulation process remains the same.

Formal proof (sketch). We prove $L_*(\Gamma) = L(G)$. First, we prove that the verification process of the simulation is valid and cannot be disturbed.

Claim 9. The verification process of simulated NonContextFree(P) in Γ is valid and cannot be disturbed by other rules.

Proof. This proof is based on Claim 2. Observe that the modified verification process requires a different sequence of rules. Consider two matching parts $01(01)^k00(01)^l01$ and $10(10)^l00(10)^k10$, where $k,l \geq 0$ and $k+l+3=m$. The process starts with $01(01)^k00(01)^l0110(10)^l00(10)^k10$. First, we use $l+1$ times the following sequence of rules: once $11 \to 00$, once $0000 \to 2222$, and four times $2 \to \varepsilon$, respectively. Next, we use rule $0000 \to 2222$ and four times rule $2 \to \varepsilon$, which verify the match of both parts. And lastly, we repeat $k+1$ times the first sequence of rules again to erase the rest.

It can be easily seen that the new nonterminal 2 cannot disturb the verification process in any way since it cannot generate anything new. It only further blocks the verification process until it is erased.

Thus, Claim 9 holds. ■

Next, we prove $L(G) \subseteq L_*(\Gamma)$; more precisely, by induction on $i \geq 0$, we demonstrate Claim 10. For brevity, let $u \Longrightarrow^l v$ denote the sequence $u \Rightarrow_{k_1} v_1 \Rightarrow_{k_2} \ldots \Rightarrow_{k_l} v_l = v$, $k_j \in \{H,I\}$, $1 \leq j \leq l$.

Claim 10. For every $w \in (N \cup T)^*$ and $i \geq 0$, $S \Rightarrow^i w$ in G implies $S \Rightarrow^*_{k_1} w_1 \Rightarrow^*_{k_2} \ldots \Rightarrow^*_{k_l} w_l = w$, $l \geq 1$, $k_j \in \{H,I\}$, $1 \leq j \leq l$, in Γ.

Proof. This proof by induction is almost identical to the proof of Claim 3. Therefore, we omit the similar parts and only present the different simulation for the non-context-free derivation step $\alpha \Rightarrow \beta$ in G. By the induction hypothesis, $S \Rightarrow^*_{k_1} w_1 \Rightarrow^*_{k_2} \ldots \Rightarrow^*_{k_l} w_l = \alpha$, $l \geq 1$, $k_j \in \{H,I\}$, $1 \leq j \leq l$, in Γ. Let $AB \to CD \in P$, $\alpha = xABy$, $\beta = xCDy$, $x,y \in (N \cup T)^*$, $A,B,C,D \in N$. From Claim 9,

$$\begin{aligned} xABy &\Rightarrow_H xCg(AB \to CD)By \\ &\Rightarrow_H xCg(AB \to CD)\,\mathrm{rev}(g(AB \to CD))Dy \\ &\Longrightarrow^{6m-1} xCDy \end{aligned}$$

in Γ. Consequently, $S \Rightarrow^*_{k_1} w_1 \Rightarrow^*_{k_2} \ldots \Rightarrow^*_{k_l} w_l = xCDy = \beta$, $l \geq 1$, $k_j \in \{H,I\}$, $1 \leq j \leq l$, in Γ.
Thus, Claim 10 holds. ■

Lastly, we prove $L_*(\Gamma) \subseteq L(G)$. Based on Claim 9, it is rather easy to demonstrate (again, a rigorous version of this demonstration is left to the reader) that Γ can generate every $y \in L_*(\Gamma)$ as

$$\begin{aligned} S = v_{0_3} &\Rightarrow^*_H v_{1_0} \Rightarrow_H v_{1_1} \Rightarrow_H v_{1_2} \Longrightarrow^{6m-1} v_{1_3} \\ &\Rightarrow^*_H v_{2_0} \Rightarrow_H v_{2_1} \Rightarrow_H v_{2_2} \Longrightarrow^{6m-1} v_{2_3} \\ &\vdots \\ &\Rightarrow^*_H v_{k_0} \Rightarrow_H v_{k_1} \Rightarrow_H v_{k_2} \Longrightarrow^{6m-1} v_{k_3} \Rightarrow^*_H v_{(k+1)_0} = y \end{aligned}$$

where for $i = 0,1,\ldots,k$ in $v_{i_3} \Rightarrow^*_H v_{(i+1)_0}$ every sentential form is over $(N \cup T)^*$; and for $j = 1,\ldots,k$ the derivation $v_{j_0} \Rightarrow_H v_{j_1} \Rightarrow_H v_{j_2} \Longrightarrow^{6m-1} v_{j_3}$ has the following form:

$$\begin{aligned} &v_{j_0} = u_jA_jB_jw_j, \\ &v_{j_1} = u_jC_jg(A_jB_j \to C_jD_j)B_jw_j, \\ &v_{j_2} = u_jC_jg(A_jB_j \to C_jD_j)\,\mathrm{rev}(g(A_jB_j \to C_jD_j))D_jw_j, \\ &v_{j_2} \Longrightarrow^{6m-1} v_{j_3} \text{ is made by using } 11 \to 00,\ 0000 \to 2222, \textit{ and } 2 \to \varepsilon, \end{aligned}$$

$v_{j_3} = u_j C_j D_j w_j$
for some $A_j B_j \to C_j D_j \in P$, $u_j, w_j \in (N \cup T)^*$.

From the derivation of the above form in Γ, we see that $v_{0_3} \Rightarrow^* v_{(k+1)_0}$ in G. Therefore, $y \in L_*(\Gamma)$ implies $y \in L(G)$. Thus, $L_*(\Gamma) \subseteq L(G)$, so $L_*(\Gamma) = L(G)$ and Theorem 8 holds. □

Corollary 11. *The resulting two-component rule-homogeneous CD grammar system Γ from the proof of Theorem 8 closely simulates the original grammar G in Kuroda normal form.*

Proof. The reasoning is the same as for Corollary 4. For any resulting Γ, we can find a bounded constant k such that for every possible derivation $u \Rightarrow v$ in G there is a k_i-step derivation in Γ that gives the same result and $k_i \leq k$. Furthermore, for a given Γ, we can easily determine the minimal possible k.

Again, any context-free rule can be simulated in one derivation step. The non-context-free rules require two initial derivation steps and the following verification process. The length of the verification is dependent on the selected size m for the verification code, and in this case it takes $6m-1$ steps to complete. The minimal possible k for a given Γ is therefore $6m+1$. □

Lastly, we show that this more restricted variant also properly works in the t mode.

Theorem 12. *Let $G = (N,T,P,S)$ be a grammar. Then, there exists a two-component rule-homogeneous CD grammar system, $\Gamma = (N',T,H,I,S)$, such that $I = \{11 \to 00,\ 0000 \to 2222\}$ and $L_t(\Gamma) = L(G)$.*

Proof.
Construction. The process of construction remains identical to Theorem 8. For a grammar $G = (N,T,P,S)$, some $m \geq 3$, and injection g, we construct the two-component rule-homogeneous CD grammar system, $\Gamma = (N',T,H,I,S)$, where $N' = N \cup \{0,1,2\}$, and H and I contain the rules as described previously.

The basic idea behind the proof remains the same as in Theorem 5. However, we have to adapt our claims for the different verification process.

Formal proof (sketch). We prove $L_t(\Gamma) = L(G)$. For convenience and brevity, consider the homomorphism $\varphi : (N' \cup T)^* \to (N \cup T)^*$ where $\varphi(a) = a$ and $\varphi(b) = \varepsilon$, for all $a \in (N \cup T)$ and $b \in \{0,1,2\}$; and let $u \Longrightarrow^t v$ denote the sequence $u \Rightarrow^t_{k_1} v_1 \Rightarrow^t_{k_2} \ldots \Rightarrow^t_{k_l} v_l = v$, $l \geq 1$, $k_j \in \{H,I\}$, $1 \leq j \leq l$.

Claim 13. For every $u \in (N \cup T)^*$ and $i \geq 0$, $S \Rightarrow^i u$ in G implies $S \Rightarrow^*_H w \Longrightarrow^t u$ in Γ, where $w \in (N' \cup T)^*$ and $\varphi(w) = u$. Furthermore, we consider w to be generally in the form $w = p_1 q_1 \ldots p_n q_n$, where $n \geq 1$, $p_j \in (N \cup T)^*$, $q_j \in \{0,1,2\}^*$, $1 \leq j \leq n$, and every q_j represents a string that can be successfully verified and erased by the verification process.

Proof. The proof by induction is analogical to Claim 6. ■

Consider $S \Rightarrow^* y$, where $y \in T^*$, in G. By Claim 13, this implies $S \Rightarrow^*_H w \Longrightarrow^t y$, where $w \in (T \cup \{0,1,2\})^*$, in Γ. It is again obvious that, in such a case, $\Rightarrow^*_H$ behaves exactly the same as $\Rightarrow^t_H$. Thus, $L(G) \subseteq L_t(\Gamma)$. It is clear that Γ working in the t mode can no longer closely simulate G. Furthermore, we even cannot bound the number how many times Γ changes its components during the generation of a sentence, since verification sequences can be arbitrarily nested and the verification process requires the constant switching of components.

The proof for $L_t(\Gamma) \subseteq L(G)$ is just a variation of the proof for $L_*(\Gamma) \subseteq L(G)$ from Theorem 8 with the modified derivation order and required string forms from Claim 13; thus, it is left to the reader.

As $L(G) \subseteq L_t(\Gamma)$ and $L_t(\Gamma) \subseteq L(G)$, $L_t(\Gamma) = L(G)$. Thus, Theorem 12 holds. □

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] Erzsébet Csuhaj-Varjú, Tomáš Masopust & György Vaszil (2009): *Cooperating distributed grammar systems with permitting grammars as components*. Romanian Journal of Information Science and Technology 12(2), pp. 175–189.

[2] Erzsebet Csuhaj-Varju, Josef Kelemen, Gheorghe Paun & Jurgen Dassow (1994): *Grammar Systems: A Grammatical Approach to Distribution and Cooperation*. Gordon and Breach Science Publishers, Inc.

[3] Filip Goldefus, Tomáš Masopust & Alexander Meduna (2010): *Left-forbidding Cooperating Distributed Grammar Systems*. Theoretical Computer Science 411(40–42), pp. 3661–3667, doi:10.1016/j.tcs.2010.06.010.

[4] Zbyněk Křivka & Tomáš Masopust (2011): *Cooperating Distributed Grammar Systems with Random Context Grammars as Components*. Acta Cybernetica 20, pp. 269–283, doi:10.14232/actacyb.20.2.2011.4.

[5] Alexander Meduna (2000): *Automata and Languages: Theory and Applications*. Springer, London.

[6] Alexander Meduna & Dušan Kolář (2002): *Homogenous Grammars with a Reduced Number of Non-Context-Free Productions*. Information Processing Letters 81(5), pp. 253–257, doi:10.1016/s0020-0190(01)00224-1.

[7] Grzegorz Rozenberg & Arto Salomaa (1997): *Handbook of Formal Languages, Vol. 1: Word, Language, Grammar*. Springer-Verlag.

[8] Grzegorz Rozenberg & Arto Salomaa (1997): *Handbook of Formal Languages, Vol. 2: Linear Modeling: Background and Application*. Springer-Verlag.

[9] Walter J. Savitch (1973): *How to Make Arbitrary Grammars Look Like Context-Free Grammars*. SIAM Journal on Computing 2(3), pp. 174–182, doi:10.1137/0202014.

[10] Derick Wood (1987): *Theory of Computation: A Primer*. Addison-Wesley, Boston.