Technical Report 2004/4,
Theoretical Computer Science Research Group, MTA SZTAKI, Budapest, 2004.

CD Grammar Systems with Competence Based Entry Conditions in Their Cooperation Protocols *

Erzsébet Csuhaj -Varjú
Computer and Automation Research Institute
Hungarian Academy of Sciences
Kende u. 13-17, H-1111 Budapest, Hungary
E-mail: csuhaj@sztaki.hu

Jürgen Dassow
Otto-von-Guericke-Universität Magdeburg
Fakultät für Informatik
PSF 4120, D-39016 Magdeburg, Germany
E-mail: dassow@iws.cs-uni.magdeburg.de

Markus Holzer
Institute für Informatik
Technische Universität München
Arcisstrasse 21, D-80290 München, Germany
E-mail: holzer@in.tum.de

Abstract

In this paper we examine context-free cooperating distributed (CD) grammar systems where the cooperation protocol is based on the competence (capability) of the component grammars in rewriting. We study the power of a derivation mode where every component is allowed to start the generation only if it has a prescribed level of competence and it is allowed to finish the work if it is not competent anymore. The competence level of a component on a string is the number of different nonterminal occurrences in this word that can be rewritten by the production set of the grammar. We show that if the prescribed level of competence of the grammar to start the derivation is equal to k or is at least k, for some natural number $k \geq 2$, then these CD grammar systems as powerful as the $ET0L$ systems with random context conditions. If this competence level is exactly one, or at least one, or it is at most k, where $k \geq 2$, then the class of $ET0L$ languages is determined by these constructions.

*The work of the first two authors was supported by a research cooperation, performed in the frame of the Hungarian-German Intergovernmental S&T Cooperation Programme, supported by the Office of Research and Development Division of the Hungarian Ministry of Education and its foreign contractual partner, BMBF. The work of the third author was supported by project Centre of Excellence in Information Technology, Computer Science and Control, ICA1-CT-2000-70025, HUN-TING project, workpackage 5.

1 Introduction

Cooperating distributed grammar systems (CD grammar systems, for short) are distributed models of language which were motivated by the syntactic properties of the blackboard architectures known from the theory of cooperative distributed problem solving [5]. A blackboard architecture consists of several autonomous agents which jointly solve a problem in turn, in such way that the agents have access to a global database, called the blackboard, which stores information on the actual state of the problem solution and the problem solving process. The problem is solved by modifying the contents of the blackboard step by step. Furthermore, the blackboard is the only mean of communication among the agents. A cooperating distributed grammar system is a construction, where several grammars jointly generate words of a language, in turn, in such way that any moment of time exactly one grammar performs a derivation step on the actual sentential form. This grammar is chosen according to the cooperation protocol of the grammars in the system, to the so-called derivation mode or cooperation strategy. According to this model, the grammars correspond to the agents, the sentential form in generation corresponds to the blackboard, and the generated language represents the set of possible problem solutions.

The idea of cooperating grammars dates back to 1978, when Meersman and Rozenberg [12], motivated by the theory of two-level grammars, introduced this term, but the theory has only been extensively and intensively explored after then Csuhaj-Varjú and Dassow [5] introduced the notion in a more general form namely, as a cooperating/distributed grammar system, and related that to the above concepts of distributed artificial intelligence, to the blackboard architectures. The interested reader can find further information in [7] and [10]. An on-line annotated bibliography on the area can be found at [8], see http://www.sztaki.hu/mms/bib.html.

Since the beginnings, cooperation protocols based on the competence (capability) of the component grammars in rewriting have outstanding role in the theory. A grammar is said to be competent on a string, if it is able to rewrite at least one nonterminal occurrence in it, and thus, the competence level of a grammar on a string is the number of different nonterminal occurrences in this word that can be rewritten by its production set.

For example, in [12] the cooperation protocol in the CD grammar system is defined as follows: a grammar is allowed to start with the derivation only if it is able to rewrite any nonterminal occurrence in the generated string, that is, the component is fully competent on the word, and it has to stop with the work if it does not have this property anymore. Later, this cooperation protocol was called sf-mode of derivation [1]. These CD grammar systems with context-free components determine the class of programmed languages with appearance checking.

According to the cooperation protocol in [5], the grammar is allowed to start the generation if it is competent on the string and it has to continue the derivation as long as it has this property. These context-free CD grammar systems prove to be essentially less powerful than the previous ones, they generate the class of $ET0L$ languages. This derivation mode is called t-derivation (terminal mode of derivation) and it is one of the most extensively investigated cooperation protocols.

Continuing this line of investigations, in [6] the power of a derivation mode, the so-called max-mode of derivation, is examined where the active grammar is always a one with the highest competence among the other components and it has to continue the derivation until and unless it does not loose this property. The generative power of these systems is between the power of the two previous variants of CD grammar systems, namely, they define a class of languages included in the class of languages of $ET0L$ systems with random context conditions. A series of papers, [2], [3], and [4] discussed cooperation protocols, where the grammars can start with the derivation if they have a prescribed level of competence and loose the right for continuing the derivation if they do not have the property anymore. These are the CD grammar systems with $(= k, comp)$-mode, $(\leq k, comp)$-mode, and $(\geq k, comp)$-modes of derivation. For example, in [2] it is shown that if the prescribed level of competence is exactly 2, then these CD grammar systems are as powerful as the CD grammar systems with sf-mode of derivations, that is, they generate the class of programmed language with appearance checking. That is, even a small level of prescribed competence leads to the same power as the demand of full competence. In [3], however, it is proved that if the prescribed level of competence is given as an upper or a lower bound, then, for competence level 2, the class of languages of random context $ET0L$ systems is defined by these systems. This class of languages is included in the class of programmed languages with appearance checking, but the problem of the properness of the inclusion is still open.

As a continuation of the previous works, the power of a derivation mode is investigated in this paper where a component is allowed to start the generation only if it has a prescribed level of competence and it is allowed to finish the work if it is not competent anymore. It is shown, that if the prescribed level of competence of the grammar to start the derivation is equal to k or is at least k, for $k \geq 2$, then these CD grammar systems are as powerful as the $ET0L$ systems with random context conditions. But, if this competence level is exactly 1, or at least 1, or at most k for $k \geq 2$, then the class of $ET0L$ languages is determined by these constructions. Notice that the case when the competence level of the grammar on the string is at least one when it starts with the derivation and finishes the work when it is no competent on the string in generation anymore is exactly the working mode by t-derivation.

2 Basic definitions

Throughout the paper we assume that the reader is familiar with formal language theory. For further information consult [7, 11, 14].

The set of nonempty words over an alphabet V is denoted by V^+, if the empty string, λ, is included, then we use notation V^*. A set of strings $L \subseteq V^*$ is said to be a language over V.

For a string $w \in V^*$, we denote the length of w by $|w|$, and for a set of symbols $U \subseteq V$ we denote by $|w|_U$ the number of occurrences of letters U in w.

For a finite language L, the number of strings in L is denoted by $card(L)$.

We specify a context-free grammar by $G = (N, T, P, S)$, where N is the set of nonterminals, T is the set of terminals, P is the set of context-free productions and S is the start symbol. We use the notation $dom(P)$ for the set $\{A \in N \mid$ there is a production $A \to \alpha \in P\}$.

We also will refer to the notion of an $ET0L$ system. An $ET0L$ system is an $n+3$-tuple $G = (N, T, P_1, \ldots, P_n, w)$, with $n \geq 1$, where N and T are defined as in the case of context-free grammars, that is, the set of nonterminals and terminals, $w \in (N \cup T)^*$ is the axiom (the initial word), and P_i, for $1 \leq i \leq n$, is a complete set of context-free productions over $(N \cup T)^*$. This means that for any symbol $X \in (N \cup T)$, the production set P_i has a rule with X being on its left-hand side. The direct derivation in an $ET0L$ system G is defined as follows: For two strings $x = x_1 \ldots x_r$ and $y = y_1 \ldots, y_r$, with $r \geq 1$, where $x_i \in (N \cup T), y_i \in (N \cup T)^*$, $1 \leq i \leq r$, we say that x directly derives y, denoted by $x \Longrightarrow_G y$, if $x_i \to y_i \in P_j$ holds for $1 \leq i \leq r$, for some j, $1 \leq j \leq r$.

By an $ET0L$ system with random context conditions or a random context $ET0L$ system, in short, we mean an $n+3$-tuple $G = (N, T, Q_1 : P_1, \ldots, Q_n : P_n, w)$, with $n \geq 1$, where N, T, w, and P_i, $1 \leq i \leq n$, are defined in the same way as for usual $ET0L$ systems, and Q_i is a finite (possibly empty) set of symbols from $(N \cup T)$, called the random context condition associated to the table P_i, for $1 \leq i \leq n$. The direct derivation in a random context $ET0L$ system is defined in the same way as for usual $ET0L$ systems, except that a table P_i, for $1 \leq i \leq n$, can be applied in a derivation step $x \Longrightarrow_G y$, if and only if each symbol of Q_i has at least one occurrence in x. If Q_i is the empty set, then no context check is necessary; in this case we can omit the indication of the context condition.

If no confusion can arise, we can omit the subscript G from the above notations $\Longrightarrow_G$.

For a grammar or a system G, of the above types, $L(G)$ denotes the language generated by G.

In the following we shall introduce the notion of a context-free CD grammar system where the components cooperate according to a derivation strategy which is based on the competence level of the component grammars in rewriting, related to the current string in generation. We first need an auxiliary notion from [6].

Definition 2.1 *Let $G = (N, T, P, S)$ be a context-free grammar and let $w \in (N \cup T)^*$. We say that production set P is of competence level k on w, $k \geq 0$, if $|dom(P) \cap alph_N(w)| = k$ holds.*

Throughout, we use notation $clev(P, w) = k$ to denote that P is with competence level k on w.

In other words, production set P is of competence level k on w if there are exactly k different nonterminals of G with an occurrence in w such that these symbols can be rewritten by a production in P. If $clev(P, w) \geq 1$, then we say P is *competent* on w. If $k = 0$, then either the production set is not competent on the string having at least one nonterminal occurrence or the string is a terminal word (including the empty word).

Now we define the notion of a context-free CD grammar system. We give the definition in a slightly different form from that can be found in [7] or in [10], since these CD grammar systems may use words longer than one as axioms. The reason of defining the concept in this way is to provide a sufficiently convenient starting mechanism which is also consistent with the customary definitions of the different variants of $ET0L$ systems. We note that both in the case of t-derivations and in the case of sf-mode of derivations the generative power of context-free CD grammar systems does not change if the system may start from a longer axiom than a symbol.

By a *context-free CD grammar system* we mean an $n+3$-tuple $\Gamma = (N, T, P_1, \ldots, P_n, w)$, with $n \geq 1$, where N, T, w are the set of nonterminals, the set of terminals and the axiom, as in the case of $ET0L$ systems, and P_i, with $1 \leq i \leq n$, are finite sets of context-free productions over $(N \cup T)$, called the components of the system. Observe that the quadruple $G_i = (N, T, P_i, w)$, for $1 \leq i \leq n$, with N, T, P_i, w, as above, can be considered as a context-free grammar with word as axiom therefore we can also speak about a component grammar or a grammar of the CD grammar system.

For two sentential forms, u and v over $(N \cup T)^*$, we say v is directly derived (derived) from u in Γ denoted by $u \Longrightarrow_\Gamma v$ ($u \Longrightarrow_\Gamma^* v$), if there is a component P_i, $1 \leq i \leq n$, such v is generated from u by a direct derivation step (by a derivation) using production set P_i.

Now we define the cooperation protocol where the component grammar starts its work if it has a prescribed level of competence and finishes the derivation if it is not competent on the string in generation anymore.

Definition 2.2 *Let* $\Gamma = (N, T, P_1, \ldots P_n, w)$, *withn* ≥ 1, *be a context-free CD grammar system and let* x, y *be two sentential forms over* $(N \cup T)^*$. *For* $k \geq 1$, *we say that* y *is directly derived from* x *in* Γ *in the* $(=k, t)$*-mode of derivation, denoted by* $x \overset{(=k,t)}{\Longrightarrow}_\Gamma y$, *if the following conditions hold: there is a component* P_i *in* Γ, *with* $1 \leq i \leq n$, *such that*

1. $clev(P_i, x) = k$ *and*
2. $clev(P_i, y) = 0$ *or, in other words, there is no word* $z \in (N \cup T)^*$ *such that* z *can be directly derived from* y *in* P_i *by applying a rule of* P_i.

We say that the above direct derivation step is a $(\leq k, t)$*-mode of direct derivation step or a* $(\geq k, t)$*- mode of direct derivation step, respectively, if condition (1) is modified as* $clev(P_i, x) \leq k$ *or* $clev(P_i, x) \geq k$, *respectively.*

For $f \in \{= k, \leq k, \geq k \mid k \geq 1\}$, *we denote by* $\overset{(f,t)}{\Longrightarrow}_\Gamma^*$ *the transitive reflexive closure of* $\overset{(f,t)}{\Longrightarrow}_\Gamma$. *If no confusion can arise, then we can omit* Γ *from the previous notations.*

Definition 2.3 *Let* $\Gamma = (N, T, P_1, \ldots P_n, w)$, *with* $n \geq 1$, *be a context-free CD grammar system.*

The language $L_{(f,t)}(\Gamma)$, *called the language of* Γ *in the* (f, t)*- mode of derivation, for* $f \in \{= k, \leq k, \geq k \mid k \geq 1\}$, *is defined as follows:*

$$L_{(f,t)}(\Gamma) = \{u \in T^* \mid w \overset{(f,t)}{\Longrightarrow}_\Gamma^* u\}.$$

That is, the language of an above type of CD grammar systems consists of those terminal words which, after starting the derivation from the axiom, can be obtained by an (f,t)-mode of derivation. If there is no component with competence level f on w, then the generated language is empty.

To help the reader in understanding, we show an example.

Example 2.1 *Let* $\Gamma = (\{S, X, X', Y, Y'\}, \{a, b, c\}, P_1, P_2, P_3, S)$ *be a CD grammar system, with*

$$P_1 = \{S \to XY, X' \to aXb, Y' \to Yc\},$$
$$P_2 = \{X \to X', Y \to Y'\},$$
$$P_3 = \{X \to ab, Y \to c\}.$$

Then, $L_{(=2,t)}(\Gamma) = L_{(\geq 2,t)}(\Gamma) = L_{(\leq 2,t)}(\Gamma) = \{a^n b^n c^n \mid n \geq 1\}$. *But* $L_{(=3,t)}(\Gamma) = L_{(\geq 3,t)}(\Gamma) = \emptyset$, *however* $L_{(\leq 3,t)}(\Gamma) = \{a^n b^n c^n \mid n \geq 1\}$.

Before turning to the results, we introduce some notations.

We denote the class of context-free languages and the class of $ET0L$ languages by $\mathcal{L}(CF)$ and $\mathcal{L}(ET0L)$. The class of languages of random context $ET0L$ systems without λ-rules is denoted as $\mathcal{L}(RC, ET0L)$, if λ-rules are allowed then we denote the corresponding language class by $\mathcal{L}(RC, ET0L, \lambda)$. If in the statement we would like to refer to both cases, then we use notation $\mathcal{L}(RC, ET0L, [\lambda])$.

Similarly, for $f \in \{= k, \leq k, \geq k \mid k \geq 1\}$, we denote by $\mathcal{L}_{(f,t)}(CF)$ the class of languages generated by context-free CD grammar systems with components without λ-rules in the (f,t)-mode of derivations. If the λ-rules are allowed, then the notation of the language class is $\mathcal{L}_{(f,t)}(CF, \lambda)$, and if we would like to refer to both cases, then we write $\mathcal{L}_{(f,t)}(CF, [\lambda])$.

It is known by the literature that $\mathcal{L}(CF) \subset \mathcal{L}(ET0L) \subset \mathcal{L}(RC, ET0L, \lambda)$. Moreover, the λ-rules have no relevance in the case of context-free grammars and that of $ET0L$ systems, the generated classes of languages are the same with and without λ-rules.

3 Results

We study the generative capacity of CD grammar systems working under the (f,t)-mode of derivations, where $f \in \{= k, \leq k, \geq k\}$ for $k \geq 1$. We prove that for derivation modes $= k$ and $\geq k$ with $k \geq 2$ these CD grammar classes determine the class of languages generated by random context $ET0L$ systems, while in the case of derivation modes $f' \in \{= 1, \geq 1\} \cup \{\leq k \mid k \geq 1\}$ a significantly less generative power, namely the generative power of $ET0L$ systems can be obtained.

Lemma 3.1

1. $\mathcal{L}_{(f,t)}(CF, [\lambda]) \subseteq \mathcal{L}(RC, ET0L, [\lambda])$, *where* $f \in \{= k, \geq k \mid k \geq 1\}$;

2. $\mathcal{L}_{(\leq k,t)}(CF, [\lambda]) \subseteq \mathcal{L}(ET0L)$, *where* $k \geq 1$.

Proof. We start with the case of $(=k,t)$-derivations where $k \geq 1$.

Let $\Gamma = (N, T, P_1, \ldots, P_n, w)$ be a CD grammar systems working in the $(=k,t)$-mode of derivation, where $k \geq 1$. We construct an $(RC, ET0L, [\lambda])$ system $G = (N', T, Q_1 : H_1, \ldots, Q_r : H_r, w)$, with $r \geq 1$, such that $L_{(=k,t)}(\Gamma) = L(G)$ holds.

G is defined as follows. To help the legibility, we list only the tables of G.

Let us define for every letter $A \in (N \cup T)$, every set of symbols M with $card(M) = k$ and $M \subseteq dom(P_i)$, $1 \leq i \leq n$, new symbols (A, i, M). For a word $w = x_1 \ldots x_m$, with $x_i \in (N \cup T)$, $1 \leq i \leq m$, let $(w, i, M) = (x_1, i, M) \ldots, (x_m, i, M)$ and let $(\lambda, i, M) = \lambda$.

Let us define tables

$H_{P_i,M,1} = M : \{A \to (A, i, M) \mid A \in (M \cup T \cup (N \setminus dom(P_i)))\} \cup \{A \to F \mid A \in (dom(P_i) \setminus M)\}$,

$H_{P_i,M,2} = \{(A, i, M) \to (w, i, M) \mid A \in N, A \to w \in P_i\} \cup \{(B, i, M) \to (B, i, M) \mid B \in (N \cup T)\}$, and, finally

$H_{P_i,M,3} = \{(A, i, M) \to A \mid A \in T \cup (N \setminus dom(P_i))\} \cup \{(A, i, M) \to F \mid A \in dom(P_i)\}$.

We show that any derivation in Γ can be simulated with a derivation in G. Let v be a sentential form generated in Γ and let us suppose that a component of Γ just finished the derivation by obtaining v. Then, either v is a terminal word, or to obtain a terminal word, some of the components, say P_i, $1 \leq i \leq n$, must continue the derivation. But this is possible if and only if there are exactly k elements in $dom(P_i)$ which have an occurrence in v. But this condition holds if and only if for some M, with $M \subseteq dom(P_i)$, $card(M) = k$, table $H_{P_i,M,1}$ can be applied to v and the resulted string, v', does not contain any occurrence of the trap symbol, F. Suppose that this is the case, that is, the application of table $H_{P_i,M,1}$ resulted in sentential form v' without any occurrence of F. Then, the derivation in G continues with the subsequent application of table $H_{P_i,M,2}$ which corresponds to a derivation in Γ performed by P_i. In CD grammar system Γ, component P_i stops with the derivation if it has no more productions applicable to the sentential form. This takes place exactly in the case when table $H_{P_i,M,3}$ can be applied to a sentential form without introducing an occurrence of F. Thus, we can see that the subsequent application of tables $H_{P_i,M,1}$, then $H_{P_i,M,2}$ several times and, finally, $H_{P_i,M,3}$ simulates a $(=k,t)$ derivation performed by component P_i in Γ.

Moreover, it is also easy to see that any sentential form over $(N \cup T)$ in G is a sentential form in Γ and reversely. (The axiom, w, is the same for both generative mechanisms.) Thus, Γ and G generate the same language. Notice that if the language of Γ is the empty set, then it is the language of G as well.

For the case of $(\geq k,t)$-derivations, the result can be obtained by replacing table $H_{P_i,M,1}$ with table $H'_{P_i,M,1} = M : \{A \to (A, i, M) \mid A \in (N \cup T)\}$.

For the case of $(\leq k,t)$ derivations, we can obtain the result by modifying table $H_{P_i,M,1}$ as follows:

$H'_{P_i,M,1} = \{A \to (A, i, M) \mid A \in (M \cup T \cup (N \setminus dom(P_i)))\} \cup \{A \to F \mid A \in (dom(P_i) \setminus M)\}$.

The reader can observe that if the CD grammar system Γ has λ-rules, then the random context $ET0L$ system or the $ET0L$ system G has λ-rules as well, otherwise both systems are without λ-rules. ■

Now we prove that random context $ET0L$ systems can be simulated with CD grammar systems with λ-rules using the $(= k, t)$-mode of derivations and the $(\geq k, t)$-mode of derivations for $k \geq 2$.

Lemma 3.2
$$\mathcal{L}(RC, ET0L, [\lambda]) \subseteq \mathcal{L}_{(f,t)}(CF, \lambda),$$
where $f \in \{= k, \geq k \mid k \geq 2\}$.

Proof. As in the previous case, we start with the case of $(= k, t)$-derivations; we prove first the statement for $k = 2$. Let $G = (N, T, Q_1 : H_1, \ldots, Q_n : H_n, w)$, where $n \geq 2$, be a random context $ET0L$ system.

Without the loss of generality we may assume that $Q_j \subseteq N$, and any production in H_j, $1 \leq j \leq n-1$, is over N and for $Q_n : H_n$ it holds that $Q_n = \emptyset$ and H_n is the set of productions $A \to a$, with $A \in N$ and $a \in T$.

(That is, no terminal symbol appears in any table H_j, $1 \leq j \leq n-1$; it is the last table H_n which introduces the terminal symbols.) Suppose that G is of the above form.

Moreover, we also may assume without the loss of generality that $Q_j \neq \emptyset$ for $1 \leq j \leq n-1$. If this is not the case for some j, then we add tables $\{A\} : H_j$ for any nonterminal letter A to the set of tables.

Now we construct a CD grammar system $\Gamma = (N', T, P_1, \ldots, P_r, w')$, $r \geq 1$, such that $L(G) = L_{(=k,t)}(\Gamma)$. The idea of the construction of Γ is as follows.

For each table $Q_i : H_i$ of G, where $1 \leq i \leq n$, and for each $M \subseteq (N \cup T)$ with $Q_i \subseteq M$, we shall define a group of components $\mathcal{P}_{M,i}$ of Γ. These grammars are dedicated to simulate the application of table $Q_i : H_i$ to sentential forms v with $alph(v) = M$.

Moreover, for any terminating derivation $w = u_1 \Longrightarrow \ldots \Longrightarrow u_n = u \in T^*$ in G, where $n \geq 1$, the simulating derivation in Γ will be of the form $w' = BwC = Bu_1C \Longrightarrow^* \ldots \Longrightarrow^* Bu_nC \Longrightarrow u_n = u$, that is, the sentential forms u_j, $1 \leq j \leq n-1$, in G correspond to sentential forms Bu_jC in Γ, where B and C new letters not in $(N \cup T)$.

Now we construct the components of Γ. First, let a for each pair (Q_i, M), with $1 \leq i \leq n$, and $M \subseteq (N \cup T)$, defined above, with $card(M) = s_{i,M}$, let us define new letters $(B, i, M, 1), \ldots, (B, i, M, 2s_{i,M})$.

Let $M = \{A_1, \ldots, A_{s_{i,M}}\}$, and without the loss of generality we may assume that $A_1, \ldots, A_j$, are the letters being elements of Q_i, where $1 \leq j \leq s_{i,M}$.

Now, let us define components in $\mathcal{P}_{M,i}$ as follows.

Let $P_{1,M,i}$ have the following rules:

$B \to (B, i, M, 1)$, $A_1 \to (A, i, M, 1)$, and $X \to F$, for any letter from $((N \cup T) \setminus M)$.

(This component grammar is for checking whether or not symbol A_1 appears in the sentential form and also checks whether or not the sentential form is over alphabet M. Notice that it is not guaranteed that any letter from M occurs in the sentential form.)

Then, for $j = 2, \ldots, s_{i,M}$ we define $P_{j,M,i}$ with rules $(B, i, M, j-1) \to (B, i, M, j)$, $A_j \to (A, i, M, j)$.

(These components are for checking whether or not letters $A_2, \ldots, A_{s_{i,M}}$ from M appear in the sentential form.)

For $l = s_{i,M} + 1, \ldots, 2s_{i,M}$, we define $P_{l,M,i}$ with rules $(B, i, M, l-1) \to (B, i, M, l)$ and $(A, i, M, l - s_{i,M}) \to \alpha$, for $A_{l-s_{i,M}} \to \alpha$ in $Q_i : H_i$.

Finally, let $P_{l,M,2s_{i,M}}$ with rules $(B, i, M, 2s_{i,M}) \to B)$ and $(A, i, M, 2s_{i,M}) \to \alpha$, for $A_{l-si,M} \to \alpha$ in $Q_i : H_i$.

(These last components simulate the application of table H_i.)

To remove the marker symbols from the sentential form and to check whether the other letter occurrences in it are only terminal letters, we define a dedicated component, P_{fin} with rules $B \to \lambda$, $C \to \lambda$, and $A \to F$, for any letter $A \in N$.

Now we show that any terminating derivation in G can be simulated by a terminating derivation in Γ. Suppose that we would like to apply table $Q_i : H_i$, $1 \leq i \leq n$, to a sentential form v at some stage of a derivation in G. The application is successful is any letter from Q_i appears in v, and then, by using productions of H_i, any letter in v is rewritten in parallel.

This derivation step will be simulated in Γ as follows. First, we guess that $alph(v) = M$ and then we consider the dedicated group of components $\mathcal{P}_{i,M}$. Obviously, if we guess that v is over another alphabet, say, M_1, then we turn to the group of productions, $\mathcal{P}_{i,M_1}$. These groups of productions are constructed in such way that any letter from M but not more has to occur in v, thus, the cases where the alphabets have common symbols are separated.

Moreover, $Q_i \subseteq M$ holds by definition, thus condition Q_i of table H_i need not to be separately checked under the simulation.

Now, the first component in $\mathcal{P}_{i,M}$, $P_{1,M,i}$ checks if the first letter of M appears in the sentential form and no other letter from $(N \cup T) \setminus M$ has an occurrence in it. This is done in such way that this grammar can start with the derivation (it is =2-competent on the sentential form) only in this case. Then, the grammar rewrites marker symbol B and also any occurrence of the previously mentioned nonterminal, say A_1, to be an indexed version.

After this, only components $P_{j,M,i}$, for $j = 2, \ldots, s_{i,M}$, where $card(M) = s_{i,M}$, can follow in succession, and they rewrite the indexed version of marker symbol B onto a corresponding indexed version (they realize a counting) and at the same time they rewrite the so far non-indexed versions of the next corresponding nonterminals from M onto their indexed versions.

If any of these grammars cannot start with the derivation, then the simulation aborts and this means that our guess $M = alph(v)$ was wrong.

But if these grammars successfully finish their work, then we obtain a sentential form of the form DvC, where D and v' are indexed versions of B and v, proving that $alph(v) = M$, and $Q_i \subseteq M$, that is, table $Q_i : H_i$ can be applied to v.

Then, the derivation in Γ continues by the work of components $P_{l,M,i}$, where $l = s_{i,M} + 1, \ldots, 2s_{i,M}$ in succession. These grammars rewrite the indexed versions of letters from $(N \cup T)$ onto words over $(N \cup T)$ that correspond to the right-hand side of the productions in H_i, and modify the indexed version of marker symbol B. The grammar in the group activated last time rewrites the indexed version of B

onto B and simulates the application of the corresponding productions.

Under the above phase of the derivation no components of Γ from other groups can be activated, thus the successful derivation performed by elements of $\mathcal{P}_{i,M}$ corresponds to a correct simulation of the application of table $Q_i : H_i$ to a sentential form v with $alph(v) = M$.

The reader can observe, that the above procedure can be repeated, by using the same or different groups of components of Γ, thus derivations in G can be correctly simulated by derivations in Γ. By the construction of the grammars of Γ, we also can see that any derivation in Γ corresponds to a derivation in G and reversely.

To finish the derivation, component P_{fin} has to be active. But this is possible if and only if the only two nonterminals in the sentential form are B and C.

Thus, we proved that $L_{(=2,t)}(\Gamma) = L(G)$.

The above construction gives a proof for the case $(\geq 2, t)$ as well, thus the statement holds for $k = 2$.

For the case of $k \geq 3$, we make the following modifications: we add further markers to increase the competence level of the components from 2 to k. That is, instead of markers B and C, we use markers $B_1, \ldots B_{k-1}$ at the beginning, that is, the axiom of G will have the form $B_1 B_2 \ldots B_{k-1} w C$. Then, we modify the other components according to this change, that is, we also take into considerations the indexed versions the markers $B_1, \ldots, B_{k-1}$. The modified proof works for the $(= k, t)$ and for the $(\geq k, t)$ derivations for $k \geq 3$. ■

By the previous two results we obtain the following theorem.

Theorem 3.1

$$\mathcal{L}(RC, ET0L, \lambda) = \mathcal{L}_{(f,t)}(CF, \lambda),$$

where $f \in \{= k, \geq k \mid k \geq 2\}$.

Now we prove that for the case of $k \geq 1$ CD grammar systems using the $(\leq k, t)$-mode of derivations determine the class of $ET0L$ languages. Moreover, this is the language class of CD grammar systems working with the $(= 1, t)$-mode of derivations or the $(\geq 1, t)$-mode of derivations as well.

Theorem 3.2

$$\mathcal{L}(ET0L) = \mathcal{L}_{(f,t)}(CF, [\lambda]),$$

where $f \in \{= 1, \geq 1\} \cup \{\leq k \mid k \geq 1\}$.

Proof. We first start with the case of $(\geq 1, t)$ derivations.

By definition, $\mathcal{L}_{(\geq 1,t)}(CF, [\lambda]) = \mathcal{L}_t(CF, [\lambda])$, thus $\mathcal{L}_{(\geq 1,t)}(CF, [\lambda]) = \mathcal{L}(ET0L)$. Now, we prove that the generated language class does not change if consider $(= 1, t)$-mode of derivations or $(\leq 1, t)$-mode of derivations. Notice that it is sufficient to give the proof for the case of $(= 1, t)$-mode of derivation, by definition, the statement follows for $(\leq 1, t)$-mode of derivations as well.

We first show that inclusion $\mathcal{L}_{(=1,t)}(CF, [\lambda]) \subseteq \mathcal{L}(ET0L)$ holds. Let, for $n \geq 1$, $\Gamma = (N, T, P_1, \ldots, P_n, w)$ be a CD grammar system working in the $(= 1, t)$-mode of derivation. We construct an $ET0L$ system $G = (N', T, H_1, \ldots, H_r, w')$, with

$r \geq 1$, such that $L_{(=1,t)}(\Gamma) = L(G)$ holds. The proof is analogous to the proof of Lemma 3.1.

G is constructed as follows. To help the legibility, again, we list only the tables of G. Let us define for every symbol $A \in (N \cup T))$, $1 \leq i \leq n$, a new symbol (A, i), and let us denote for $w = x_1 \ldots x_m$, with $x_i \in (N \cup T)$, where $1 \leq i \leq m$, $(w, i) = (x_1, i) \ldots, (x_m, i)$ and let $(\lambda, i) = \lambda$. Let F be a new nonterminal, the trap symbol.

Let us define tables

$H_{P_i,A,1} = \{A \to (A, i)\} \cup \{D \to (D, i) \mid D \in ((N \cup T) \setminus dom(P_i))\} \cup \{B \to F \mid B \in (dom(P_i) \setminus \{A\})\}$,

$H_{P_i,A,2} = \{(A, i) \to (w, i) \mid A \in N, A \to w \in P_i\} \cup \{(B, i) \to (B, i) \mid B \in (N \cup T)\}$, and, finally

$H_{P_i,A,3} = \{(A, i) \to A \mid A \in T \cup (N \setminus dom(P_i))\} \cup \{(A, i) \to F \mid A \in dom(P_i)\}$.

Using analogous arguments to the proof of Lemma 3.1., it is easy to see that the application of table $H_{P_i,A,1}$ without introducing the trap symbol, F, corresponds to the case when component P_i is exactly 1-competent on a sentential form, with nonterminal A providing the competence. Then, we also can easily see, that the application of table $H_{P_i,A,2}$ corresponds to a derivation performed by P_i, continuing the one that started before, and, finally, the application of table $H_{P_i,A,3}$ without introducing the trap symbol means that P_i stopped with the derivation after performing a t-derivation. Similarly to the argumentation used in the proof of Lemma 3.1, we can show that $L_{(=1,t)}(\Gamma) = L(G)$ holds. Obviously, the same proof can be used for proving the statement for the case of $(\leq 1, t)$-derivations.

Now we prove that the reverse inclusion, that is, $\mathcal{L}(ET0L) \subseteq \mathcal{L}_{(=1,t)}(CF, [\lambda])$ holds. The proof is based on similar considerations as the proof of Lemma 3.2.

Let $G = (N, T, H_1, \ldots, H_n, w)$, with $n \geq 1$, be an $ET0L$ system. We construct a CD grammar system $\Gamma = (N', T, P_1, \ldots, P_r, w')$, with $r \geq 1$, such that $L_{(=1,t)}(\Gamma) = L(G)$ holds.

For each table H_i, $1 \leq i \leq n$, and for each subset M of $(N \cup T)$, we shall construct a group of components $\mathcal{P}_{i,M}$ of Γ which is dedicated to simulate the application of table H_i to a sentential form v with $alph(v) = M$. For this reason, we introduce new letters (A, i, M) and $(A, i, M)'$ for each letter $A \in (N \cup T)$, each table H_i, $1 \leq i \leq n$, and for each $M \subseteq (N \cup T)$. Moreover, for any table H_i and any set M, defined as above, we introduce new marker symbols (B, i, M).

Now we construct the components in $\mathcal{P}_{i,M}$ as follows. Let us assume that $M = \{A_1, \ldots, A_r\}$, $r \geq 1$. First, let $P_{i,M,0}$ with the only production $B \to (B, i, M)$. (This component indicates that we simulate the application of table H_i to a sentential form over alphabet M.

Then, for $1 \leq j \leq r$, where $card(M) = r$, we set

$P_{i,M,j}$ with productions $A_j \to (A_j, i, M)$, $(A_l, i, M) \to F$, for $l > j$, $1 \leq l \leq n$, $A_h \to F$, $1 \leq h < j$, $B \to F$,and (B, j, M') for $1 \leq j \neq i \leq n$, and $M' \neq M$, with $M' \subseteq N$.

(These components rewrite the letters occurring in the sentential form to their indexed version; the rewriting is possible only if the components follow the order of symbols $A_1, \ldots, A_r$, and for each symbol A_r there is at least one occurrence in the sentential form.)

The next component in $\mathcal{P}_{i,M}$ is $P_{i,M,c}$ with production $(B,i,M) \to (B,i,M)'$.

(This component indicates that the previous "colouring" procedure has been finished, and the simulation of the application of the productions will follow.)

To do this, we define a series of components as follows: for $1 \leq j \leq r$ we set

$P'_{i,M,j}$ with productions $(A_j, i, M \to \alpha)$, where $A_j \to \alpha \in H_i$, and α is the primed version of α, and $(A_l, i, M) \to F$, for $l > j$.

(These components rewrite the indexed versions of the symbols according to the productions in table H_i.)

Finally, there is a component $P_{i,M,fin}$ with the only production $(B,i,M)' \to B$.

(This component resets the marker symbol, B.)

In order to guarantee the correct simulation of the $ET0L$ system, we define the axiom w' of the CD grammar system Γ as $w' = wB$, and we add a further component P_{fin} with productions $B \to F$ and $X \to F$ for any letter X which is not a terminal symbol.

We can easily see that the terminating derivations in Γ simulate the terminating derivations in G and only that. The derivation in Γ can start with the work of component $P_{0,i,M}$ for some i, $1 \leq i \leq n$, which by introducing the marker symbol (B,i,M) indicates that the simulation of the application of table H_i follows. Suppose now that the current sentential form is $v(B,i,M)$, where $v \in (N \cup T)^*$. Then, production sets $P_{i,M,j}$ must follow each other, checking whether $alph(v) = M$ holds. If no trap symbol is introduced, then the marker symbol (B,i,M) is changed for $(B,i,M)'$, and the next series of components, $P'_{i,M,j}$ rewrite the indexed versions of the letters in M according to the corresponding productions in H_i. These production sets must follow each other in succession, no other productions set can be active during this phase of the derivation without introducing the trap symbol, F. After the end of this derivation phase, the marker symbol will be reset to B, and the procedure is repeated as many times as necessary to obtain a terminal word with production set P_{fin}. The components of Γ were defined in such way that only the derivations described above lead to terminal words. Thus, Γ and G determine the same language. Hence, we proved the result.

The reader can notice that the proof of the inclusion $\mathcal{L}(ET0L) \subseteq \mathcal{L}_{(=1,t)}(CF,[\lambda])$ above is a proof for the inclusion $\mathcal{L}(ET0L) \subseteq \mathcal{L}_{(\leq k,t)}(CF,[\lambda])$, for $k \geq 1$, as well, since Γ was constructed in such way that to obtain a terminal word the component grammars had to be of competence level $= 1$ when they started the derivation. Combining this proof with the proof of the corresponding statement of Lemma 3.1., we obtain the result. ■

References

[1] H. Bordihn, E. Csuhaj-Varjú: On competence and completeness in CD grammar systems. Acta Cybernetica 12(4) (1996), 347-361.

[2] M.H. ter Beek, E. Csuhaj-Varjú, M. Holzer, Gy. Vaszil: On competence in cooperating distributed grammar systems, Part I., Technical Report, Research Group on Modelling Multi-Agent Systems, MTA SZTAKI, 2002/1.

[3] M.H. ter Beek, E. Csuhaj-Varjú, M. Holzer, Gy. Vaszil: On competence in cooperating distributed grammar systems, Part II., Technical Report, Research Group on Modelling Multi-Agent Systems, MTA SZTAKI, 2002/2.

[4] M.H. ter Beek, E. Csuhaj-Varjú, M. Holzer, Gy. Vaszil: On competence in cooperating distributed grammar systems, Part III.,Technical Report, Research Group on Modelling Multi-Agent Systems, MTA SZTAKI, 2002/3.

[5] E. Csuhaj-Varjú, J. Dassow: On cooperating/distributed grammar systems. Journal of Information Processing and Cybernetics EIK 26(1990), 49-63.

[6] E. Csuhaj-Varjú, J. Dassow, M. Holzer: On a competence-based cooperation strategy in CD grammar systems. Technical Report, Theoretical Computer Science Research Group, MTA SZTAKI, 2004/1.

[7] E. Csuhaj-Varjú, J. Dassow, J. Kelemen, Gh. Păun: Grammar Systems - A Grammatical Approach to Distribution and Cooperation. Topics in Computer Mathematics 5. Gordon and Breach Science Publishers, Yverdon, 1994.

[8] E. Csuhaj-Varjú, Gy. Vaszil: An annotated bibliography of grammar systems. See: http://www.sztaki.hu/mms/bib.html

[9] J. Dassow, Gh. Păun: Regulated Rewriting in Formal Language Theory. EATCS Monograph on Theoretical Computer Scince 18, Springer-Verlag, Berlin-Heidelberg-New York, 1989.

[10] J. Dassow, Gh. Păun, G. Rozenberg: Grammar Systems. Handbook of Formal Languages. Vol 2., Chapter 4. (G. Rozenberg, A. Salomaa, eds.), Springer Verlag, Berlin-Heidelberg-New York, 1997, 155-213.

[11] Handbook of Formal Languages, Vol. 1-3., (G. Rozenberg, A. Salomaa, eds.), Springer-Verlag, Berlin-Heidelberg-New York, 1997.

[12] R. Meersman, G. Rozenberg: Cooperating grammar systems. In: Proc. MFCS'78, Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1978, 364-373.

[13] G. Rozenberg, A. Salomaa: The Mathematical Theory of L Systems. Academic Press, New York, 1980.

[14] A. Salomaa: Formal Languages. Academic Press, New York, 1973.