Acta Cybernetica **12** (1996) 427-444.

Teams in Grammar Systems: Hybridity and Weak Rewriting *

Maurice H. ter BEEK †

Abstract

Some new ideas in the theory of teams in grammar systems are introduced and studied. Traditionally, a team is formed from a finite number of sets of productions and in every derivation step, one production from each component is used to rewrite a symbol of the sentential form. Hence rewriting is done in parallel. Several derivation modes are considered, varying from using a team exactly one time to using it a maximal amount of times. Here, the possibility of different teams having different modes of derivation is defined, as is a weaker restriction on the application of a team. The generative power of such mechanisms is investigated.

1 Introduction

In [4], *cooperating distributed grammar systems* (CD grammar sytems for short) were introduced to formalize a link, recognized in [6], between the so-called *multi-agent systems* theory in *Artificial Intelligence* and the theory of formal languages. Since then these systems have been studied intensively and this has already resulted in the monograph [5], which contains an exhaustive survey of the state of the art in the so-called theory of grammar systems until ca. 1992.

By now, many well-motivated enhancements have been introduced, resulting in *hybrid* CD grammar systems (allowing the grammars to have different capabilities, [22]) and *team* CD grammar systems (grouping the grammars in teams and rewrite in parallel, [20]), to name but a few.

Here hybrid (prescribed) team CD grammar systems are defined, thus allowing work to be done in teams while at the same time assuming these teams to have different capabilities. Two basically different versions can be defined. One can consider a hybrid CD grammar system and automatically form teams of its components according to some strategy or one can consider a CD grammar system

*This research was supported by a scholarship from the Hungarian Ministry of Culture and Education. Moreover, the facilities provided by the Department of General Computer Science of the Eötvös Loránd University and in particular by the Computer and Automation Research Institute of the Hungarian Academy of Sciences were essential.

†Department of Computer Science, Leiden University, P.O. Box 9512, 2300 RA Leiden, The Netherlands. E-mail: mtbeek@wi.leidenuniv.nl

with prescribed teams and simply associate a (possibly different) so-called mode of derivation with each team. Concerning the latter one it will be shown that this hybridity does not enlarge the generative power any further. However, every recursively enumerable language can be generated by a hybrid prescribed team CD grammar system with teams of two members. The question whether the automatic forming of teams enlarges the generative power of hybrid CD grammar systems remains an open problem.

Furthermore, a variant of the way teams work in the literature so far is presented. The motivation to introduce a different concept of rewriting is twofold. Not only is the strict requirement that every component of the team must participate in every step often bothering in generating languages but, perhaps more important, it is definitely too restrictive in the most recent application of grammar systems as a framework for natural language generation (see, e.g., [8] and [10]).

This new way of rewriting is called *weak* rewriting and it is investigated in the case of teams in eco-grammar systems in [2]. It resembles the well-known concept of appearance checking in regulated rewriting: every component of a team which contains a production that can rewrite the sentential form must be used, but a component which does not contain any production with a left-hand side that is contained in the sentential form does not need to be used. The generative power of CD grammar systems with prescribed teams of variable size operating in the weak rewriting step will be shown to equal that of the class of programmed grammars with unconditional transfer. This implies that these families and those of the prescribed team CD grammar systems operating in the traditional rewriting step and the same modes of derivation do not coincide.

Finally, in the special case of prescribed team CD grammar systems with only one production per component and teams of variable size, an equality with the class of unordered scattered context grammars is presented. This leads to the fact that there are several cases when only one production per component suffices for prescribed team CD grammar systems with teams of variable size.

2 Preliminaries

In this section, some prerequisites necessary for understanding the sequel are defined. For details and unexplained notions, the reader is referred to [28] for formal languages, [13] for regulated rewriting, [27] for Lindenmayer systems and [5], [9], [11], [24] and [3] for (variants of) grammar systems.

The set of all non-empty strings over an *alphabet* V is denoted by V^+. If the *empty string*, λ, is included, the notation becomes V^*. The *length* of a string x is denoted by $|x|$.

An *inclusion* is denoted by $\subseteq$, whereas a *proper inclusion* is denoted by $\subset$.

Sometimes, the notation for a family of languages contains a λ between the brackets [and]. This means that the statement holds in the case of allowing λ-productions (indicated by the λ inbetween brackets) as well as in the case of a restriction to λ-free productions (thus neglecting the λ inbetween brackets). Also other symbols between brackets must now be understood.

Without definition, the family of context-free languages (CF) is used in the sequel. Its definition can be found in, e.g., [13]. The same holds for the family of languages generated by ET0L systems ($ET0L$). Finally, also the family of languages generated by [hybrid] CD grammar systems ($[H]CD$) shall not be defined here. However, their definitions can be found in [5] and will become clear in the sequel.

None of the above families of languages will be used in any construction in the proofs. Those families of languages that are used in (some of) the proofs below, are defined next.

An *unordered scattered context grammar with appearance checking* ([21]) is a construct $G = (N, T, S, P, F)$, where N is the set of nonterminals, T is the set of terminals, $S \in N$ is the axiom, $P = \{p_1, p_2, \ldots, p_n\}$ is a finite set of *rules* (rules are of the form $p_i : (\alpha_1, \alpha_2, \ldots, \alpha_{m_i}) \to (\beta_1, \beta_2, \ldots, \beta_{m_i})$, where $\alpha_j \to \beta_j$ are productions over $N \cup T$) and F is a set of occurrences of productions in P, $1 \leq i \leq n$. For $w, w' \in (N \cup T)^*$ and $1 \leq i \leq n$ it is said that w directly derives w', written as

$$\begin{aligned} w \Longrightarrow w' \quad \text{iff} \quad & w = w_1\alpha_{i_1}w_2\alpha_{i_2}\ldots w_m\alpha_{i_m}w_{m+1},\ w' = w_1\beta_{i_1}w_2\beta_{i_2}\ldots w_m\beta_{i_m}w_{m+1}, \\ & p_i : (\alpha_1, \alpha_2, \ldots, \alpha_p) \to (\beta_1, \beta_2, \ldots, \beta_p) \in P,\ (\alpha_{i_1}, \alpha_{i_2}, \ldots, \alpha_{i_m}) \text{ is a} \\ & \text{permutation of a subsequence of } (\alpha_1, \alpha_2, \ldots, \alpha_p),\ w_l \in (N \cup T)^* \\ & \text{and } 1 \leq l \leq m+1 \\ \text{and} \quad & \alpha_j \text{ in } \{\alpha_1, \alpha_2, \ldots, \alpha_p\} \text{ and not in } \{\alpha_{i_1}, \alpha_{i_2}, \ldots, \alpha_{i_m}\} \text{ implies that} \\ & \alpha_j \text{ is not contained in } w \text{ and } \alpha_j \to \beta_j \in F. \end{aligned}$$

If $F = \emptyset$, the unordered scattered context grammar is called an *unordered scattered context grammar without appearance checking* and F is omitted from the construct. Moreover, if F contains all occurrences of productions in P, the unordered scattered context grammar is called *with unconditional transfer*. The language generated by G is $L(G) = \{w \in T^* \mid S \Longrightarrow^* w\}$, where $\Longrightarrow^*$ denotes the reflexive and transitive closure of $\Longrightarrow$.

The family of languages generated by unordered scattered context grammars with λ-free context-free productions in P is denoted by USC_{ac} in the case of grammars with appearance checking; when grammars without appearance checking are considered the subscript ac is omitted and when grammars with unconditional transfer are considered the subscript ac is replaced by ut.

A *matrix grammar with appearance checking* is a construct $G = (N, T, S, M, F)$, where N is the set of nonterminals, T is the set of terminals, $S \in N$ is the axiom, M is a finite set of *matrices* of the form $m : (r_1, r_2, \ldots, r_n)$, where $r_i : \alpha_i \to \beta_i$ are productions over $N \cup T$ and $|\alpha|_N \geq 1$, $1 \leq i \leq n$ and F, finally, is a set of occurrences of productions in M. For $w, w' \in (N \cup T)^*$ and $m : (\alpha_1 \to \beta_1, \alpha_2 \to \beta_2, \ldots, \alpha_n \to \beta_n) \in M$ it is said that w directly derives w', written as

$$w \Longrightarrow w' \quad \text{iff} \quad \text{there exist } w_0, w_1, \ldots, w_n \in (N \cup T)^* \text{ such that}$$

$w_0 = w$ and $w_n = w'$ and for all $0 \le i \le n-1$

either $w_{i-1} = w'_{i-1}\alpha_i w''_{i-1}$ and $w_i = w'_{i-1}\beta_i w''_{i-1}$ for some $w'_{i-1}, w''_{i-1} \in (N \cup T)^*$

or the production $\alpha_i \to \beta_i$ cannot be applied to w_{i-1}, $\alpha_i \to \beta_i \in F$ and $w_i = w_{i-1}$.

If $F = \emptyset$, the matrix grammar is called a *matrix grammar without appearance checking* and F is omitted from the construct. Moreover, if F contains all occurrences of productions in M, the matrix grammar is called *with unconditional transfer*. The language generated by G is $L(G) = \{w \in T^* \mid S \Longrightarrow^* w\}$, where $\Longrightarrow^*$ denotes the reflexive and transitive closure of $\Longrightarrow$.

The family of languages generated by matrix grammars with λ-free context-free productions in M is denoted by MAT_{ac} in the case of grammars with appearance checking; when grammars without appearance checking are considered the subscript *ac* is omitted and when grammars with unconditional transfer are considered the subscript *ac* is replaced by *ut*.

For all generative devices mentioned above, only the notation in the case of λ-free context-free productions was given. When there is no restriction to λ-free productions a superscript λ is added to the notation.

3 Teams in grammar systems

Definition 1 *Let N and T be two disjoint alphabets. A* production *over (N,T) is a pair $(A,x) \in N \times (N \cup T)^*$. Usually, $A \to x$ shall be written instead of (A,x). If $x \neq \lambda$, then $A \to x$ is called a* λ-free *production. A* team *over (N,T) is a multiset of sets of productions over (N,T). The sets of productions occurring in a team shall be referred to as* components.

Traditionally, a team rewrites a string in the following manner. Here, this original notion is renamed *strong* rewriting since another way of rewriting is introduced after this definition.

Definition 2 *Let N and T be two disjoint alphabets. Let Q be a team over (N,T) and $x, y \in (N \cup T)^*$. Then x is rewritten by Q, in the strong rewriting step, into y, written as*

$$x \stackrel{s}{\Longrightarrow}_Q y \quad \textit{iff} \quad x = x_1A_1x_2A_2\ldots x_nA_nx_{n+1},\ y = x_1y_1x_2y_2\ldots x_ny_nx_{n+1},$$
$$x_i \in (N \cup T)^*,\ 1 \le i \le n+1,\ A_j \to y_j \in P_j,\ 1 \le j \le n \textit{ and } Q = \{P_1, P_2, \ldots, P_n\}.$$

A derivation step of a team thus consists of choosing a production from each component of this team and applying these in parallel on the string to be rewritten.

Now the *weak* rewriting step for teams is introduced. It is loosely based on the so-called weakly competitive rewriting step for colonies as introduced in [12].

Definition 3 *Let N and T be two disjoint alphabets. Let Q be a team over (N,T) and $x, y \in (N \cup T)^*$. Then x is rewritten by Q, in the weak rewriting step, into y, written as*

$$x \overset{w}{\Longrightarrow}_Q y \quad \text{iff} \quad \begin{array}{l} x = x_1A_1x_2A_2\ldots x_nA_nx_{n+1},\ y = x_1y_1x_2y_2\ldots x_ny_nx_{n+1}, \\ x_i \in (N \cup T)^*,\ 1 \le i \le n+1,\ A_j \to y_j \in P_j,\ 1 \le j \le n \text{ and} \\ \{P_1, P_2, \ldots, P_n\} \subseteq \{P_1, P_2, \ldots, P_s\} = Q \text{ such that} \\ \text{for all } P_q \in Q \setminus \{P_1, P_2, \ldots, P_n\} \text{ there exists} \\ \text{no production } \alpha \to \beta \in P_q \text{ such that } \alpha \in x_1x_2\ldots x_{n+1}. \end{array}$$

The weak rewriting step of a team thus works in the same way as the strong rewriting step, as far as choosing a production from each component of this team and applying these in parallel on the current sentential form is concerned. However, a derivation according to the strong rewriting step is blocked (1) when a component of the team does not contain a production with a left-hand side that is contained in the current sentential form or (2) when two (or more) components can only rewrite a symbol of the current sentential form that appears only once in that sentential form. In the weak rewriting step neither case results in a blocked derivation, since only every component containing a production that can rewrite a symbol from the current sentential form, without clashing with another component for wanting to rewrite the same symbol, applies these productions in parallel on the current sentential form.

If Q is a singleton team, i.e. $Q = \{P\}$ for some set of productions P, then $x \overset{-}{\Longrightarrow}_P y$ shall be written instead of $x \overset{-}{\Longrightarrow}_{\{P\}} y$, for $- \in \{s, w\}$. It is clear that in that case only one symbol in x is rewritten, using a production from P.

So-called modes of derivation are used to prescribe halting requirements on the use of a team. These modes can be divided into three groups. Firstly, mode $*$ has *no restrictions* whatsoever. Any number of derivation steps is allowed. Secondly, modes $\le k$, $= k$ and $\ge k$ restrict the number of derivation steps to *at most*, *exactly* and *at least* k derivation steps, respectively. Thirdly, modes t_0, t_1 and t_2 are modes that represent a so-called *maximal* number of derivation steps. All three prescribe a slightly different condition which needs to be fulfilled before a team is considered to have successfully worked in that mode. In the case of mode t_0 the work of a team ends successfully when *no further derivation step can be done as a team*, in the case of mode t_1 the work ends when *no component of the team can apply one of its productions any longer* and in mode t_2, finally, the work of a team ends when *there is at least one component that can no longer apply one of its productions.* For these so-called maximal derivation modes, a distinction is made between the weak and the strong rewriting step.

Definition 4 *Let $Q = \{P_1, P_2, \ldots, P_n\}$ be a team over (N,T) and let $f \in \{\le k, = k, \ge k \mid k \ge 1\} \cup \{*, t_0, t_1, t_2\}$ be a* mode (of derivation). *Furthermore, let*

$x, y, z \in (N \cup T)^*$ and $k \in \mathbf{N}$. Then x is rewritten by Q, in the weak ($- = w$) or strong ($- = s$) rewriting step and working in mode f, into y, written as

$$
\begin{array}{llll}
x & \stackrel{-}{\Longrightarrow}{}_Q^{\leq k} & y & \textit{iff} \quad x \stackrel{-}{\Longrightarrow}{}_Q^{k'} y \textit{ for some } k' \leq k, \\
x & \stackrel{-}{\Longrightarrow}{}_Q^{=k} & y & \textit{iff} \quad x \stackrel{-}{\Longrightarrow}{}_Q^{k} y, \\
x & \stackrel{-}{\Longrightarrow}{}_Q^{\geq k} & y & \textit{iff} \quad x \stackrel{-}{\Longrightarrow}{}_Q^{k'} y \textit{ for some } k' \geq k, \\
x & \stackrel{-}{\Longrightarrow}{}_Q^{*} & y & \textit{iff} \quad x \stackrel{-}{\Longrightarrow}{}_Q^{k} y \textit{ for some } k, \\
x & \stackrel{-}{\Longrightarrow}{}_Q^{t_0} & y & \textit{iff} \quad x \stackrel{-}{\Longrightarrow}{}_Q^{*} y \textit{ and there is no } z \textit{ such that } y \stackrel{-}{\Longrightarrow}_Q z, \\
x & \stackrel{s}{\Longrightarrow}{}_Q^{t_1} & y & \textit{iff} \quad x \stackrel{s}{\Longrightarrow}{}_Q^{*} y \textit{ and for no component } P_i \in Q \textit{ and no } z \\
 & & & \qquad \textit{there is a derivation } y \stackrel{s}{\Longrightarrow}_{P_i} z \textit{ and} \\
x & \stackrel{s}{\Longrightarrow}{}_Q^{t_2} & y & \textit{iff} \quad x \stackrel{s}{\Longrightarrow}{}_Q^{*} y \textit{ and there is a component } P_i \in Q \\
 & & & \qquad \textit{for which there is no derivation } y \stackrel{s}{\Longrightarrow}_{P_i} z.
\end{array}
$$

The three variants of the t-mode of derivation first appeared in [17] (t_0), [20] (t_1) and [26] (t_2); the other modes of derivation are the natural extension of the modes in CD grammar systems (see [5]) to teams of grammars.

Now a more general definition of teams in the theory of grammar systems than the original one from [20] and its generalization from [26] can be introduced.

Definition 5 *A* hybrid prescribed team CD grammar system *is a construct*

$$\Gamma = (N, T, S, P_1, P_2, \ldots, P_n, (Q_1, f_1), (Q_2, f_2), \ldots, (Q_m, f_m)),$$

where N is the set of nonterminals, T is the set of terminals, with $N \cap T = \emptyset$, $S \in N$ is the axiom, $P_1, P_2, \ldots, P_n$ are sets of productions over (N, T), $Q_1, Q_2, \ldots, Q_m$ are teams with components from $P_1, P_2, \ldots, P_n$ and $f_1, f_2, \ldots, f_m$ are modes of derivation.

If, in this construct, $f_i = f_j$ for all $1 \leq i, j \leq m$, the definition of a prescribed team CD grammar system as in [26] is obtained.

Note that in this definition, there is no restriction on the size of a team. In the original definition of teams in [20], however, they are of constant size. A natural number $s \geq 1$ is given and the teams are formed such that the number of components of every team is exactly s; these teams are called of constant size s. Moreover, in that definition the teams are not prescribed, but each set of components can be a team (so-called *free* teams) as long as the size restriction is fulfilled.

It is now clear that one can differentiate between the following four variants of teams in the theory of grammar systems. For all four, hybridity is another possibility.

Free teams of constant size: this is the original definition of [20], as explained above.

Free teams of variable size: each subset of components can be a team.

Prescribed teams of constant size: all prescribed teams consist of the same number of components.

Prescribed teams of variable size: these are defined in Definition 5.

In the case of teams of constant size, whether prescribed or free, a finite set of axioms $W \subseteq (N \cup T)^*$, with only one string in it containing nonterminals, is allowed. This is done since otherwise in the case of λ-free productions no string shorter than s could be generated. In the case of free teams with teams of constant size, the construct thus becomes $\Gamma = (N, T, W, P_1, P_2, \ldots, P_n)$. The modifications in the other cases are obvious.

Definition 6 *Consider a hybrid prescribed team CD grammar system Γ as in Definition 5. Then the language generated by Γ, operating in the weak ($- = w$) or strong ($- = s$) rewriting step, is*

$$L^-(\Gamma) = \{z \in T^* \mid S \overset{-}{\Longrightarrow}{}^{f_{i_1}}_{Q_{i_1}} w_{i_1} \overset{-}{\Longrightarrow}{}^{f_{i_2}}_{Q_{i_2}} \cdots \overset{-}{\Longrightarrow}{}^{f_{i_p}}_{Q_{i_p}} w_{i_p} = z,\ 1 \leq i_j \leq m,\ 1 \leq j \leq p\}.$$

When dealing with a language generated by teams of constant size, the notation of Definition 6 is modified to $L^-(\Gamma, s)$. When the teams are not hybrid, the mode of derivation is added as a subscript to this notation.

The family of languages generated by CD grammar systems with hybrid prescribed teams of variable size, operating in the strong rewriting step and λ-free context-free productions is denoted by HPT_*CD. When teams are of constant size s, the $*$ in the notation is replaced by s and when there is no restriction to λ-free productions, λ is added to the notation as a superscript. When the teams are not hybrid (prescribed) the H (P) in the notation is omitted.

The weak rewriting step is only considered in the sequel for CD grammar systems with prescribed teams of variable size. The family of languages generated by such systems, working in derivation mode f and operating in the weak rewriting step, is denoted by $PT_wCD(f)$ in the case of λ-free context-free productions; when λ-productions are allowed the superscript λ is added.

Instead of prescribing the hybrid teams, another way to introduce hybrid teams is defined next. Consider a hybrid CD grammar system and automatically form teams by combining all components with a certain mode of derivation to form a team with that mode of derivation. Because the teams are formed automatically, they are not part of the system "hardware", but a way to define the work of the system.

Definition 7 *Consider a hybrid CD grammar system*

$$\Gamma = (N, T, S, (P_1, f_1), (P_2, f_2), \ldots, (P_n, f_n)),$$

where N is the set of nonterminals, T is the set of terminals, with $N \cap T = \emptyset$, $S \in N$ is the axiom, $P_1, P_2, \ldots, P_n$ are sets of productions over (N,T) and $f_1, f_2, \ldots, f_m$ are modes of derivation.

Then teams $(Q_i, g_i) \subseteq \{(P_1, f_1), (P_2, f_2), \ldots, (P_n, f_n)\}$ are automatically formed in the following way. For $g_i \in \{*, t_0, t_1, t_2\} \cup \{\leq k, =k, \geq k \mid k \geq 1\}$

$$(Q_i, g_i) = \{(P_k, f_k) \mid f_k = g_i,\ 1 \leq k \leq n\}.$$

Such a team $(Q_i, g_i) = \{(P_{j_1}, f_{j_1}), (P_{j_2}, f_{j_2}), \ldots, (P_{j_s}, f_{j_{s_i}})\}$, is called an automatically formed team working in mode g_i.

The language generated by Γ with automatically formed teams is

$$L^{aut}(\Gamma) = \{z \in T^* \mid S \Longrightarrow_{Q_{i_1}}^{g_{i_1}} w_{i_1} \Longrightarrow_{Q_{i_2}}^{g_{i_1}} \cdots \Longrightarrow_{Q_{i_m}}^{g_{i_m}} w_{i_m} = z,\ m \geq 1\}.$$

The family of languages generated by hybrid CD grammar systems with automatically formed teams of variable size and only λ-free context-free productions is denoted by HT_*CD; when λ-productions are allowed the notation becomes HT_*CD^λ. Note that due to the automatical construction from a hybrid CD grammar system (with a one-symbol axiom), the notion of teams of constant size is very restricted. Only teams of constant size 1 could be constructed, but they obviously have the same generative power as the underlying hybrid CD grammar system. Naturally, it is possible to consider hybrid CD grammar systems with a string axiom instead of a single nonterminal.

Some relations concerning the generative power of several of these grammar systems discussed above are given next. A more complete overview can be found in [1]. In the first paper on teams in grammar systems, [20], it was proved that, for $f \in \{=1, \geq 1, *\} \cup \{\leq k \mid k \geq 1\}$,

$$CF = T_1CD(f) \subset T_2CD(f) \text{ and}$$
$$ET0L = T_1CD(t) \subset T_2CD(t_1).$$

These relations prove that there are modes of derivation for which the forming of teams strictly increases the power of CD grammar systems, since $CD(t) = ET0L$ and $CF = CD(=1) = CD(\geq 1) = CD(*) = CD(\leq k)$ for a $k \geq 1$ were already known to hold (see, e.g., [5]). In [7] it was proved that teams of size two suffice, i.e. for $s \geq 2$

$$T_sCD(t_1) \subseteq T_2CD(t_1).$$

The main results of [26] are, for $s \geq 2$, $f \in \{*\} \cup \{\leq k, = k, \geq k \mid k \geq 1\}$ and $g \in \{t_1, t_2\}$,

$$PR^{[\lambda]} = PT_sCD^{[\lambda]}(f) = PT_*CD^{[\lambda]}(f) \text{ and}$$
$$PR_{ac}^{[\lambda]} = T_sCD^{[\lambda]}(g) = PT_sCD^{[\lambda]}(g) = PT_*CD^{[\lambda]}(g)$$

and the main result of [17] is, for $s \geq 2$ and $h \in \{t_0, t_1\}$,

$$MAT_{ac}^{[\lambda]} = T_sCD^{[\lambda]}(h) = PT_sCD^{[\lambda]}(h) = PT_*CD^{[\lambda]}(h) = T_*CD^{[\lambda]}(h).$$

4 Homogeneous versus heterogeneous teams

The next lemma follows immediately from the definitions stated in the previous section.

Lemma 1 *For* $s \geq 1$ *and* $f \in \{*, t_0, t_1, t_2\} \cup \{\leq k, =k, \geq k \mid k \geq 1\}$

(i) $T_sCD^{[\lambda]}(f) \subseteq PT_sCD^{[\lambda]}(f) \subseteq PT_*CD^{[\lambda]}(f)$,
$T_*CD^{[\lambda]}(f) \subseteq PT_*CD^{[\lambda]}(f) \subseteq HPT_*CD^{[\lambda]}$ *and*
$PT_sCD^{[\lambda]}(f) \subseteq HPT_sCD^{[\lambda]} \subseteq HPT_*CD^{[\lambda]}$,

(ii) $HCD^{[\lambda]} = HT_1CD^{[\lambda]} \subseteq HPT_sCD^{[\lambda]} \subseteq HPT_*CD^{[\lambda]}$ *and*
$HT_1CD^{[\lambda]} \subseteq HT_*CD^{[\lambda]} \subseteq HPT_*CD^{[\lambda]}$ *and*

(iii) $[H][P]T_sCD^{[\lambda]} \subseteq [H][P]T_{s+1}CD^{[\lambda]}$.

It is natural to ask whether results similar to those that were stated in the previous section, can be obtained for the new definitions concerning hybrid teams of grammars. Indeed, some similar results for the hybrid cases will be proved below, but some open problems remain.

To begin with, some results concerning hybrid prescribed team CD grammar systems are presented. The next corollary follows immediately from Lemma 1 and results stated in the previous section.

Corollary 1 *For* $s \geq 2$

$$PR_{ac}^{[\lambda]} \subseteq HPT_sCD^{[\lambda]}.$$

For the λ-free case the next lemma is necessary to conclude that hybrid prescribed team CD grammar systems cannot generate more than the non-hybrid ones.

Lemma 2

$$HPT_*CD^{[\lambda]} \subseteq MAT_{ac}^{[\lambda]}.$$

Proof Consider the hybrid prescribed team CD grammar system

$$\Gamma = (N, T, S, P_1, P_2, \ldots, P_n, (Q_1, f_1), (Q_2, f_2), \ldots, (Q_m, f_m)).$$

Define the homomorphism h from $(N \cup T)^*$ into $(\{A' \mid A \in N\} \cup T)^*$ by

$$h(a) = a \text{ for } a \in T \text{ and } h(A) = A' \text{ for } A \in N.$$

Moreover, associate to a team $Q_i = \{P_{i_1}, P_{i_2}, \ldots, P_{i_{s_i}}\}$, $1 \leq i \leq m$, all sequences of productions such that from each component P_{i_j}, $1 \leq j \leq s_i$, exactly one production is included in such a sequence. Denote such a sequence by $\sigma = (A_1 \to x_1, \ldots, A_s \to x_s)$ and all such sequences associated to a team Q_i by $Seq_i = \{\sigma_{i_1}, \sigma_{i_2}, \ldots, \sigma_{i_{l_i}}\}$, $1 \leq i \leq m$.

To simulate this hybrid prescribed team CD grammar system, construct the following matrix grammar

$$G' = (N', T', S', M', F'),$$

where

$$\begin{aligned}
N' &= N \cup \{A' \mid A \in N\} \cup \{T, F\} \cup \{\Sigma_{i_j}, \Sigma'_{i_j} \mid 1 \leq j \leq l_i, 1 \leq i \leq m\} \cup \\
&\quad \{[Q_i, f_i, j] \mid (Q_i, f_i) \in \Gamma, f_i \in \{\leq k, = k, \geq k\}, 1 \leq i \leq m, 0 \leq j \leq k\} \cup \\
&\quad \{[Q_i, g_i], [Q_i, t_0]' \mid (Q_i, g_i) \in \Gamma, g_i \in \{*, t_0, t_1, t_2\}, 1 \leq i \leq m\}, \\
T' &= T \cup \{z\}, \\
M' &= \{(S' \to ST)\} \cup \\
&\quad \{(T \to [Q_i, f_i, 0]) \mid f_i \in \{\leq k, = k, \geq k\}, 1 \leq i \leq m\} \cup \\
&\quad \{(T \to [Q_i, g_i]) \mid g_i \in \{*, t_0, t_1, t_2\}, 1 \leq i \leq m\} \cup \\
&\quad \{([Q_i, f_i, j] \to [Q_i, f_i, j+1], A_1 \to h(x_1), A_2 \to h(x_2), \ldots, A_s \to h(x_s)) \mid \\
&\qquad 0 \leq j \leq k-1, (Q_i, f_i) = \{P_{j_1}, P_{j_2}, \ldots, P_{j_s}\}, A_r \to x_r \in P_{j_r}, \\
&\qquad f_i \in \{\leq k, = k, \geq k\}, 1 \leq i \leq m, 1 \leq r \leq s\} \cup \\
&\quad \{([Q_i, \geq k, k] \to [Q_i, \geq k, k], A_1 \to h(x_1), A_2 \to h(x_2), \ldots, A_s \to h(x_s)) \mid \\
&\qquad (Q_i, \geq k) = \{P_{j_1}, P_{j_2}, \ldots, P_{j_s}\}, A_r \to x_r \in P_{j_r}, 1 \leq i \leq m, 1 \leq r \leq s\} \cup \\
&\quad \{([Q_i, g_i] \to [Q_i, g_i], A_1 \to h(x_1), A_2 \to h(x_2), \ldots, A_s \to h(x_s)) \mid \\
&\qquad (Q_i, g_i) = \{P_{j_1}, P_{j_2}, \ldots, P_{j_s}\}, A_r \to x_r \in P_{j_r}, g_i \in \{*, t_0, t_1, t_2\}, \\
&\qquad 1 \leq i \leq m, 1 \leq r \leq s\} \cup \\
&\quad \{([Q_i, t_0] \to \Sigma_{i_1}) \mid 1 \leq i \leq m\} \cup \\
&\quad \{(\Sigma_{i_j} \to \Sigma'_{i_{j+1}}, A_1 \to \varphi_1, A_2 \to \varphi_2, \ldots, A_s \to \varphi_s) \mid \\
&\qquad \sigma_{i_j} = (A_1 \to x_1, \ldots, A_s \to x_s), \varphi_r \in \{A'_r, F\}, \varphi_r = F \text{ must hold for} \\
&\qquad \text{at least one } r, 1 \leq r \leq s, 1 \leq j \leq l_i - 1, 1 \leq i \leq m\} \cup \\
&\quad \{(\Sigma'_{i_{l_i}} \to [Q_i, t_0]', A_1 \to \varphi_1, A_2 \to \varphi_2, \ldots, A_s \to \varphi_s) \mid \\
&\qquad \sigma_{i_{l_i}} = (A_1 \to x_1, \ldots, A_s \to x_s), \varphi_r \in \{A'_r, F\}, \varphi_r = F \text{ must hold for} \\
&\qquad \text{at least one } r, 1 \leq r \leq s, 1 \leq i \leq m\} \cup \\
&\quad \{(\Sigma'_{i_j} \to \Sigma_{i_j}, A'_1 \to F, A'_2 \to F, \ldots, A'_k \to F) \mid \\
&\qquad \{A_1, A_2, \ldots, A_k\} = N, 1 \leq j \leq l_i, 1 \leq i \leq m\} \cup \\
&\quad \{(A' \to A) \mid A \in N\} \cup \\
&\quad \{([Q_i, \leq k, j] \to T), ([Q_i, = k, k] \to T), ([Q_i, \geq k, k] \to T), ([Q_i, *] \to T) \mid
\end{aligned}$$

$$1 \le i \le m, 0 \le j \le k\} \cup$$
$$\{([Q_i, t_0]' \to T, A_1' \to F, A_2' \to F, \ldots, A_k' \to F) \mid$$
$$\{A_1, A_2, \ldots, A_k\} = N, 1 \le i \le m\} \cup$$
$$\{([Q_i, t_1] \to T, A_1 \to F, A_1' \to F, A_2 \to F, A_2' \to F, \ldots, A_r' \to F) \mid$$
$$\{A_1, A_2, \ldots, A_r\} = \bigcup_{P_j \in (Q_i, t_1)} dom(P_j), 1 \le i \le m\} \cup$$
$$\{([Q_i, t_2] \to T, A_1 \to F, A_1' \to F, A_2 \to F, A_2' \to F, \ldots, A_r' \to F) \mid$$
$$\{A_1, A_2, \ldots, A_r\} = dom(P_j) \text{ for some } P_j \in (Q_i, t_2), 1 \le i \le m\} \cup$$
$$\{(T \to z)\} \text{ and}$$

in F' are all the productions $A \to F$ appearing in M'.

The simulation of Γ starts with introducing the sentential form ST, in which S is the start-symbol of Γ and T is a marker. The marker will control the derivation and S will generate the language of the hybrid CD grammar system with prescribed teams. This marker is non-deterministically replaced by a control symbol of the form $[Q_i, f_i, j]$ or $[Q_i, g_i]$. In these nonterminals, Q_i is the team working in mode f_i or g_i and j is a counter, necessary for the modes $f_i \in \{\le k, = k, \ge k\}$. With teams working in mode $g_i \in \{*, t_0, t_1, t_2\}$ we do not need to count and the third component is omitted.

When the marker $[Q_i, f_i, j]$ ($[Q_i, g_i]$) is present in the sentential form a simulation by Q_i in mode f_i (g_i) is simulated. The homomorphism h priming all nonterminals in the matrices is necessary to guarantee that the productions are applied to nonterminals that were already existing in the sentential form before these matrices were applied and not to those introduced by a production from these matrices themselves. The counter in the case of modes $\le k$, $= k$ and $\ge k$ guarantees that a team rewrites the sentential form less than k, exactly k or at least k times, respectively. In case of mode $*$, t_0, t_1 and t_2 there is no counting at all.

In case of t_1 and t_2, however, the productions in the set F guarantee that a team does not stop rewriting until no more component or at least one component of the team can no longer be used, respectively. Finally, in mode t_0 the symbol $[Q_i, t_0]$ can be replaced only by Σ_{i_1}. This symbol can then be replaced by $\Sigma'_{i_{j+1}}$ and back to $\Sigma_{i_{j+1}}$ until $\Sigma'_{i_{k_i}}$ is reached. In this way the correct termination of Q_i in mode t_0 is checked, by the following restrictions.

Firstly, Σ_{i_j} can only be replaced by $\Sigma'_{i_{j+1}}$ if the corresponding sequence of productions indeed cannot be used anymore. An F is introduced otherwise, since each sequence must have at least one $\varphi_r = F$. Secondly, $\Sigma'_{i_{j+1}}$ is allowed to be replaced by $\Sigma_{i_{j+1}}$ only after all primed symbols have been replaced by their originals. Finally, $\Sigma'_{i_{k_i}}$ can only be replaced by $[Q_i, t_0]'$ after indeed none of the sequences Σ_{i_j}, $1 \le j \le l_i$, can be used and then eventually be replaced by T.

In every case, afterwards the primes are removed and another team can non-deterministically take the marker spot and start its simulation in its mode. Eventually a terminal string results from S followed by the marker T. This marker is then replaced by z thus yielding $L(G') = L(\Gamma)\{z\}$. This symbol z can be removed by a morphism and thus, since it is known from [13] that the family MAT_{ac} is closed under restricted morphisms, $L(\Gamma) \in MAT_{ac}$ and the first statement of the lemma is proved.

$HPT_*CD^\lambda \subseteq MAT^\lambda_{ac}$ can be proved directly by a similar construction, even simplified since the marker can eventually be replaced by λ, making the use of a morphism unnecessary. □

It is known that $PR^{[\lambda]}_{ac} = MAT^{[\lambda]}_{ac}$ (see, e.g., [13]), hence the following corollary follows directly from Lemma 2.

Corollary 2 $HPT_*CD^{[\lambda]} \subseteq PR^{[\lambda]}_{ac}$.

All these results for hybrid prescribed team CD grammar systems immediately lead to a result for hybrid CD grammar systems with automatically formed teams, presented next.

Corollary 3 *For* $s \geq 1$

$$HT_1CD^{[\lambda]} \subseteq HT_*CD^{[\lambda]} \subseteq PR^{[\lambda]}_{ac}.$$

Combining these lemmas and corollaries concerning the new definitions, the following theorem is obtained.

Theorem 1 *For* $s \geq 2$

$$HT_*CD^{[\lambda]} \subseteq HPT_*CD^{[\lambda]} = HPT_sCD^{[\lambda]} = PR^{[\lambda]}_{ac}.$$

5 Weak versus strong rewriting

It is not hard to see that the principle of weak rewriting, not having to apply productions if they cannot be applied, resembles the appearance checking feature in regulated rewriting. Therefore, the following lemma does not come as a surprise. In the sequel, a restriction to only one production per component will be indicated by a 1 added as subscript. To be even more precise, denote U_mSC_{ut} for the class of unordered scattered context grammars with unconditional transfer and m scattered context rules and denote $P_mT_wCD_1(f)$ for the class of prescribed team CD grammar systems with m teams of variable size, 1 production per component, working in mode f and operating in the weak rewriting step.

Lemma 3 *For* $m \geq 1$ *and* $f \in \{=1, \geq 1, *\} \cup \{\leq k \mid k \geq 1\}$

$$U_mSC^{[\lambda]}_{ut} = P_mT_wCD^{[\lambda]}_1(f) \text{ and } U_mSC^{[\lambda]} = P_mT_*CD^{[\lambda]}_1(f).$$

Proof Only the inclusion from left to right of the first statement is proved here, all other inclusions can be proved in a similar straigthforward way. Consider an unordered scattered context grammar

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

with unconditional transfer and m scattered context rules. Moreover, for $P = \{p_1, p_2, \ldots, p_m\}$, $p_i : (\alpha_{i,1}, \alpha_{i,2}, \ldots, \alpha_{i,k_i}) \to (\beta_{i,1}, \beta_{i,2}, \ldots, \beta_{i,k_i})$ and $1 \leq i \leq m$, denote

$$r_{i,j} = \alpha_{i,j} \to \beta_{i,j} \text{ for } 1 \leq j \leq k_i.$$

To simulate this unordered scattered context grammar, construct the prescribed team CD grammar system

$$\Gamma = (N, T, S, P_1, P_2, \ldots, P_n, Q_1, Q_2, \ldots, Q_m),$$

where

$P_1, P_2, \ldots, P_n$	are	the components $\{r_{i,j}\}$ for $1 \leq j \leq k_i$ and $1 \leq i \leq m$ and
$Q_1, Q_2, \ldots, Q_m$	are	the teams $\{\{r_{1,j}\}, \{r_{2,j}\}, \ldots, \{r_{m,j}\}\}$ for $1 \leq j \leq k_i$ and $1 \leq i \leq m$.

A parallel rewriting step of an unordered scattered context grammar is simulated by a parallel rewriting step of a team, with its components being exactly the same productions as in the scattered context rule. Every component contains exactly one such a production and the number of teams equals the number of scattered context rules. Any production in G as well as in Γ does not have to be applied, if it cannot be applied to the sentential form.

Note that the proof requires the unordered character of the scattered context grammar, for a component of a team can rewrite any occurrence of the left-hand side of its production in the current sentential form. Since a team has to simulate the use of a scattered context rule, its mode of derivation is restricted to the cases as stated in the lemma. Clearly, $L(\Gamma) = L(G)$ and the lemma is proved for the case with as well as for the case without λ-productions. □

This lemma has some interesting corollaries.

Corollary 4 *For* $x \in \{s, *\}$, $f \in \{=1, \geq 1, *\} \cup \{\leq k \mid k \geq 1\}$ *and* $g \in \{*\} \cup \{\leq k, =k, \geq k \mid k \geq 1\}$

$$PR_{ut}^{[\lambda]} = PT_wCD_1^{[\lambda]}(f) \not\subseteq PT_xCD^{[\lambda]}(g).$$

Proof The equalities $PR_{ut}^{[\lambda]} = USC_{ut}^{[\lambda]}$ can be found in [16] and Lemma 3 thus leads to the equality in the statement. In [19] it is proved that the language

$\{a^{2^n} \mid n \geq 1\}$ cannot be generated by $PR^{[\lambda]}$. However, the programmed grammar (with unconditional transfer)

$$G_1 = (\{S, A, F\}, \{a\}, S, P),$$

where

$$\begin{aligned} P \;=\; & \{(1 : S \to AA, \{1,2,5\}, \{1,2,5\}), \\ & (2 : S \to F, \{3\}, \{3\}), \\ & (3 : A \to S, \{3,4\}, \{3,4\}), \\ & (4 : A \to F, \{1\}, \{1\}), \\ & (5 : A \to a, \{5\}, \{5\})\} \end{aligned}$$

generates $L(G_1) = \{a^{2^n} \mid n \geq 1\} \in PR_{ut}^{[\lambda]}$ and thus $PR_{ut}^{[\lambda]} \not\subseteq PR^{[\lambda]}$ holds. Finally, $PR^{[\lambda]} = PT_xCD^{[\lambda]}(g)$, for $x \in \{s, *\}$ and $g \in \{*\} \cup \{\leq k, =k, \geq k \mid k \geq 1$, is stated in Section 3. □

Thus, for several modes of derivation, a prescribed team CD grammar system with only 1 production per component and operating in the weak rewriting mode cannot be simulated by a prescribed team CD grammar system operating in the strong rewriting step not even when there is no limit of 1 production per component.

Corollary 5 *For* $f \in \{=1, \geq 1, *\} \cup \{\leq k \mid k \geq 1\}$

$$CD(t) \subset PT_wCD_1(f) \subset PT_wCD_1^{\lambda}(f).$$

Proof The equality $CD(t) = ET0L$ can be found in [5]. The strict inclusion $ET0L \subset O$, where O denotes the family of languages generated by the ordered grammars (with context-free productions) as introduced in [18], can be found in [13]. Furthermore, $O \subset PR_{ut}$ can be found in [14]. In [16], $PR_{ut} = USC_{ut}$ is proved. Finally, in [15], it was proved that $PR_{ut} \subset PR_{ut}^{\lambda}$. Together with Lemma 3 these results lead to a proof of the statement. □

Hence, for several modes of derivation, already a prescribed team CD grammar system with only 1 production per component and operating in the weak rewriting step can generate more than a CD grammar system working in mode t can.

Corollary 6 *For* $f \in \{=1, \geq 1, *\} \cup \{\leq k \mid k \geq 1\}$

$$PT_*CD^{[\lambda]}(f) = PT_*CD_1^{[\lambda]}(f).$$

Proof These results follow from Lemma 3 and the fact that $USC^{[\lambda]} = PR^{[\lambda]}$ (see, e.g., [13]) and $PR^{[\lambda]} = PT_*CD^{[\lambda]}(f)$ for $f \in \{*\} \cup \{\leq k, =k, \geq k \mid k \geq 1\}$ (see

Section 3) hold. □

Hence teams with one production per component suffice for prescribed team CD grammar systems with teams of variable size operating in derivation mode $=1$, ≥ 1, $*$ or $\leq k$ (for a $k \geq 1$).

Remark 1 *Note that $CD(f) = CF$ (see Section 3), though $CF \subset PT_*CD_1(f)$ (see Section 3 and Corollary 6), for $f \in \{=1, \geq 1, *\} \cup \{\leq k \mid k \geq 1\}$. Hence even CD grammar systems with n components cannot generate all languages that can be generated by prescribed team CD grammar systems with teams of variable size and only 1 production per component, for modes $f \in \{=1, \geq 1, *\} \cup \{\leq k \mid k \geq 1\}$.*

6 Open problems

It is clear that many open problems remain, both in the field of homogeneous versus heterogeneous teams as in the case of weak versus strong rewriting. To start with the latter: is strong rewriting more powerful than weak rewriting, or is the class of programmed grammars with unconditional transfer equal to the class of programmed grammars with appearance checking? My conjecture is the former, since the latter would settle the conjecture $PR_{ut}^{[\lambda]} \subset PR_{ac}^{[\lambda]}$ in the negative and this very interesting open problem in the theory of formal languages is very widely conjectured to hold. In fact, in [29], the class of programmed grammars is claimed to be closed under intersection with regular sets (which would result in a proper inclusion indeed), but the proof is subject to disbelief (see, e.g., [15]).

A possible angle into solving this open problem is to investigate the generative power of prescribed team CD grammar systems operating in the weak rewriting step with a maximal derivation mode. This might help to fill or to definitely establish the gap between programmed grammars with unconditional transfer and those with appearance checking. More investigation into the weak rewriting step might also finally prove $PR^{[\lambda]} \not\subseteq PR_{ut}^{[\lambda]}$.

It is interesting to note that also for colonies (for a definition of colonies, see, e.g., [12]) and for teams in eco-grammar systems ([2]), the relation between weak and strong rewriting is unknown. An answer to those relations would not necessarily solve the case for teams in CD grammar systems, but it might shed light on some intrinsic characteristics of weak versus strong rewriting. However, in the case of colonies no relation between the two ways of rewriting is known yet, whereas in the case of eco-grammar systems it was proved in [2] that strong rewriting can be simulated by weak rewriting.

Concerning homogeneous and heterogeneous teams, the main open problem is whether automatic forming of teams strictly increases the generative power of hybrid CD grammar systems. The conjecture, at least for the λ-free case, is yes since this would result in confirmation of the conjecture, stated in [23], that the inclusion $HCD \subseteq MAT_{ac}$ is proper. This might be a difficult open problem to settle since several years after their introduction in [22] still many problems concerning hybrid CD grammar systems are open.

Especially the relation with matrix grammars is wide open, since in [23] also the relation between matrix grammars without appearance checking and hybrid CD grammar systems is posed as an open problem. However, several different angles have been provided so far. For example, in [1], graph controlled hybrid CD grammar systems ($GCHCD$) were defined and they were proved to be included in the matrix grammars with appearance checking and to include both the hybrid CD grammar systems and the matrix grammars without appearance checking. It is not known, however, whether these inclusions are proper or whether equalities can be proved, but one of the inclusions of $MAT \subseteq GCHCD \subseteq MAT_{ac}$ must be proper. A solution to (one of) these open problems could shed light on this relation between hybrid CD grammar systems and matrix grammars without appearance checking, or perhaps even solve this open problem.

Acknowledgements

This work has benefited from discussions with and comments and suggestions from E. Csuhaj-Varjú, H.C.M. Kleijn and Gh. Păun. This paper is an excerpt from Part III: *Teams in CD grammar systems* of my master's thesis ([1]).

References

[1] M. H. ter Beek, Teams in grammar systems, *IR-96-32 (master's thesis)*, Leiden University, 1996.

[2] M. H. ter Beek, Simple eco-grammar systems with prescribed teams. To appear in *Grammatical Models of Multi-Agent Systems*, Gordon and Breach, London, 1997.

[3] M. H. ter Beek, Teams in grammar systems: sub-context-free cases. To appear in *Developments in Regulated Rewriting and Grammar Systems, Lecture Notes in Computer Science* (1997).

[4] E. Csuhaj-Varjú and J. Dassow, On cooperating distributed grammar systems. *J. Inf. Process. Cybern. EIK* 26 (1990), 49 - 63.

[5] E. Csuhaj-Varjú, J. Dassow, J. Kelemen and Gh. Păun, *Grammar Systems. A Grammatical Approach to Distribution and Cooperation*, Gordon and Breach, London, 1994.

[6] E. Csuhaj-Varjú and J. Kelemen, Cooperating grammar systems: a syntactical framework for the blackboard model of problem solving. In *Proc. AI and information-control systems of robots '89* (I. Plander, ed.), North-Holland Publ. , 1989, 121 - 127.

[7] E. Csuhaj-Varjú and Gh. Păun, Limiting the size of teams in cooperating grammar systems. *Bulletin EATCS* 51 (1993), 198 - 202.

[8] E. Csuhaj-Varjú, Grammar systems: a framework for natural language generation. In *Mathematical Aspects of Natural and Formal Languages (Gh. Păun, ed.), World Scientific Series in Computer Science* 43 (1994), World Scientific, Singapore, 63 - 78.

[9] E. Csuhaj-Varjú, Eco-grammar systems: recent results and perspectives. In [25] (1995), 79 - 103.

[10] E. Csuhaj-Varjú, Generalized eco-grammar systems: a framework for natural language generation. In *Lenguajes Naturales Y Lenguajes Formales XII (C. Martin-Vide,ed.)*, PPU, Barcelona, 1996, 13-27.

[11] J. Dassow, Cooperating grammar systems (definitions, basic results, open problems). In [25] (1995), 40 - 52.

[12] J. Dassow, J. Kelemen and Gh. Păun, On parallelism in colonies. *Cybernet. Systems* 24 (1993), 37 - 49.

[13] J. Dassow and Gh. Păun, *Regulated Rewriting in Formal Language Theory*, Springer-Verlag, 1989.

[14] H. Fernau, Membership for 1-limited ET0L languages is not decidable. *J. Inform. Process. Cybern. EIK* 30 (1994), 191 - 211.

[15] H. Fernau, On unconditional transfer. *Proceedings of the MFCS'96, Lecture Notes in Computer Science* 1113, Springer-Verlag, Berlin, 1996, 348 - 359.

[16] H. Fernau, Scattered context grammars with regulation. *Ann. Univ. Bucureşti, Math-Informatics Series* 45, 1 (1996), 41 - 50.

[17] R. Freund and Gh. Păun, A variant of team cooperation in grammar systems. *J. UCS* 1, 2 (1995), 105 - 130.
http://hyperg.iicm.tu-graz.ac.at

[18] I. Friš, Grammars with partial ordering of the rules. *Inform. Control* 12 (1968), 412 - 425. Correction in *Inform. Control* 14 (1969), 5.

[19] D. Hauschildt and M. Jantzen, Petri net algorithms in the theory of matrix grammars, *Acta Informatica* 31 (1994), 719 - 728.

[20] L. Kari, A. Mateescu, Gh. Păun and A. Salomaa, Teams in cooperating grammar systems, *J. Exper. Th. AI* 7 (1995), 347 - 359.

[21] O. Mayer, Some restricted devices for context-free languages. *Inform. Control* 20 (1972), 69 - 92.

[22] V. Mitrana, Hybrid cooperating distributed grammar systems. *Computers and AI* 2 (1993), 83 - 88.

[23] Gh. Păun, On the generative capacity of hybrid CD grammar systems, *J. Inform. Process. Cybern. EIK* 30, 4 (1994), 231 - 244.

[24] Gh. Păun, Grammar systems: a grammatical approach to distribution and cooperation. In *Automata, Languages and Programming; 22nd International Colloquium, ICALP'95, Szeged, Hungary, Lecture Notes in Computer Science* 944 (1995), 429 - 443.

[25] *Artificial Life: Grammatical Models* (Gh. Păun, ed.), Black Sea Univ. Press, Bucharest, Romania, 1995.

[26] Gh. Păun and G. Rozenberg, Prescribed teams of grammars. *Acta Informatica* 31 (1994), 525 - 537.

[27] G. Rozenberg and A. Salomaa, *The Mathematical Theory of L Systems*, Academic Press, New York, 1980.

[28] A. Salomaa, *Formal Languages*, Academic Press, New York, 1973.

[29] E. D. Stotskii, Control of the conclusion in formal grammars. *Problems of Information Transmission* 7, 3 (1971, translated 1973), 257 - 270.