App. Math. and Comp. Intel., Vol. 1 (2012) 12–23
http://amci.unimap.edu.my

Descriptional complexity of lindenmayer systems

S. Turaev*, **G. Mavlankulov, M. Othman, and M. H. Selamat**

Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
43400 UPM Serdang, Selangor, Malaysia.

Received: 29 February 2012; Revised: 17 July 2012; Accepted: 18 July 2012

Abstract: *In this paper we study the nonterminal complexity of Lindenmayer systems with respect to tree controlled grammars. We show that all 0L, D0L and E0L languages can be generated by tree controlled grammars with at most five nonterminals. The results based on the idea of using a tree controlled grammar in the t-normal form, which has the one active nonterminal, and a coding homomorphism.*

Keywords: *Context-free languages, L systems, Tree controlled grammars, Descriptional complexity, Nonterminal complexity.*

PACS: *02.70.-c, 89.20.Ff*

---

1 Introduction

Formal language theory, introduced by Chomsky in the 1950s as a tool for a description of natural languages [1, 2, 3], has also been widely involved in modeling and investigating phenomena appearing in Computer Science and other related fields. The symbolic representation of a modeled system in the form of strings makes its processes by information processing tools very easy. Coding Theory, Cryptography, Computation Theory, and many other fields use sets of strings for the description and analysis of modeled systems. In the modeling, we usually have to deal with infinite sets of strings with respect to the number of symbols. Thus, it is natural to define some finite devices which generate these types of set of strings. If we consider strings as *words*, similarly in natural languages, then a set of strings can be considered as a (formal) *language* and a generative device as a *grammar*. A grammar generally consists of finite sets of *terminal* and *nonterminal* symbols, a finite set of *production rules* and the *axiom*. A derivation of a word starts with the axiom and in each step, some subword of an obtained word are replaced by another subword using production rules until the word is produced.

With respect to the forms of production rules, grammars can be divided into two major classes: *context-free* (exactly one nonterminal symbol can be replaced) and *context-sensitive* (a subword containing at least one nonterminal can be replaced). Context-free grammars

*Corresponding Author: sherzod@fsktm.upm.edu.my (S. Turaev)

have beautiful mathematical properties and are easily applicable in practical problems. They were first used in the study of human languages. More modern examples include the structures of mark-up languages like HTML and XML. Another important application of context-free grammars occurs in the specification and compilation of programming languages. Context-free grammars improve the process of implementing parsers into a routine job that can be completed very quickly. However, they cannot cover all aspects of modeled phenomena. On the other hand, context-sensitive grammars are too powerful to be used in applications, and they have bad features, for instance, for context-sensitive grammars, the emptiness problem is undecidable and the existing algorithms for the membership problem, thus for the parsing, have exponential complexities. In order to overcome this problem, we need some "context-free like" generative devices, which have as many context-free like properties as possible, but are also able to describe the non-context-free features of the specific languages in question. One of the solutions is that a context-free grammar should be considered with some additional (control) mechanism which restricts the application of the rules in order to avoid some derivations and obtain a subset of the language generated in usual way. The computational power of some grammars with control mechanism turns out to be greater than the power of context-free grammars. The consideration of different types of control mechanisms leads to the definition of different types of grammars with controlled (regulated) rewriting. In the monograph [4], we can find the detailed information on various types of grammars with regulated rewriting such as matrix, programmed, valence, random context, tree controlled grammars, etc.

The notion of a *tree controlled grammar* was introduced in [5] as a regulated generative device which is a very simple and natural extension of context-free grammars. A tree controlled grammar is defined as a context-free grammar with some regular language where the structure of the derivation trees is restricted by requirement that all words belonging to a level of the derivation tree have to be an element of the regular language. The investigation of tree controlled grammars is interesting for many reasons, for instance, each word generated by a tree controlled grammar there is a derivation tree exactly like in the context-free case, and tree controlled grammars are considerably more powerful (Turing complete) than context-free grammars, yet for a large class of languages generated by tree controlled grammars parsing methods working in quadratic time are available.

Though the descriptions of grammars (the number of nonterminal symbols and production rules) for languages are finite, they may increase with respect to the number of terminal symbols. Thus, it is always important to study language generative mechanisms from the point of view of the *descriptional complexity*: the number of nonterminals and the number of the production rules.

The study of the descriptional complexity with respect to regulated grammars started in [6, 7, 8, 9, 10]. In recent years several interesting results on this topic have been obtained. There are results which compare the conciseness of minimal descriptions of languages by different types of regulated grammars as well as statements that grammars with a bounded size suffice to generate all languages of certain language classes. For instance, the nonterminal complexity of programmed and matrix grammars is studied in [11], where it is shown that three nonterminals for programmed grammars with appearance checking, and four nonterminals for matrix grammars with appearance checking are enough to generate every recursively enumerable language. There are several papers which present analogous results for scattered context grammars [12, 13, 14, 15] and semi-conditional grammars [15, 16, 17].

The study of the descriptional complexity of tree controlled grammars was started in [18], where a number of preliminary results on the nonterminal complexity of tree controlled grammars were obtained, particularly, it was shown that tree controlled grammars with no more than nine nonterminals are enough to generate all recursively enumerable languages.

This bound was improved to seven in [19]. Moreover, it was shown that all linear and regular simple matrix languages can be generated by tree controlled grammars with the nonterminal complexity bounded by three. In [20] the optimality of this bound was proved, and also showed that tree controlled grammars with the nonterminal complexity bounded by four are sufficient to generate all context-free languages. In this paper we start the study of the nonterminal complexity of tree controlled grammars generating *parallel* languages, i.e., languages generated by *Lindenmayer* systems. As an initial result, we show that five nonterminals are sufficient for tree controlled grammars to generate all 0L, D0L (deterministic 0L), and E0L (extended 0L) languages.

The paper is organized as follows. In Section 2 we give some notions and definitions from the theory of formal languages needed in sequel. In Section 3 we define a normal form for tree controlled grammars, called *the t-normal form*, and show that for every E0L system, one can construct equivalent tree controlled grammars in the t-normal form. In Section 4 we prove that every L language can be generated by a tree controlled grammar with no more than five nonterminals.

2 Preliminaries

We assume that the reader is familiar with the basic notations of formal language theory, for details refer to [4, 21].

Let T be an *alphabet* which is a finite nonempty set of symbols. A *string* over the alphabet T is a finite sequence of symbols from T. The *empty* string is denoted by ε. The set of all strings over the alphabet T is denoted by T^*. A subset of T^* is called a *language*.

A *context-free grammar* is specified as a quadruple $G = (N, T, P, S)$ where N and T are the disjoint alphabets of terminals and nonterminals, respectively, $P \subseteq N \times (N \cup T)^*$ is a finite set of context-free productions, and S is the axiom. Usually, a rule $(u, v) \in P$ is written in the form $u \to v$. A rule of the form $u \to \varepsilon$ is called an *erasing rule*. A grammar is called *regular* if $P \subseteq T^*N \cup T^*$ (i.e., if all its rules are of the form $A \to wB$ or $A \to w$ with $A, B \in N$ and $w \in T^*$).

By $\mathrm{Var}(G)$ we denote the number of the nonterminals of a grammar G, i.e.,

$$\mathrm{Var}(G) = |N|.$$

A string $x \in (N \cup T)^*$ *directly derives* a string $y \in (N \cup T)^*$ in G, written as $x \Rightarrow y$ if and only if there is a rule $A \to v \in P$ such that $x = x_1Ax_2$ and $y = x_1vx_2$ for some $x_1, x_2 \in (N \cup T)^*$. The reflexive and transitive closure of the relation $\Rightarrow$ is denoted by $\Rightarrow^*$. The *language* generated by G, denoted by $L(G)$, is defined by

$$L(G) = \{w \in T^* \mid S \Rightarrow^* w\}.$$

With each derivation in a context-free grammar G, one can associate a derivation tree. The *level* associated with a node is the number of edges in the path from the root to the node. The *height* of the tree is the largest level number of any node. With a derivation tree t of height k and each number $0 \le i \le k$, we associate the *word of level i* which is given by all nodes of level i read from left to right, and we associate the *sentential form of level i* which consists of all nodes of level i and all leaves of level less than i read from left to right. Obviously, if u and v are sentential forms of two successive levels, then $u \Rightarrow^* v$ holds and this derivation is obtained by a parallel replacement of all nonterminals occurring in the sentential form u.

A *tree controlled grammar* is defined as a quintuple $H = (N, T, P, S, R)$ where $G = (N, T, P, S)$ is a context-free grammar and $R \subseteq (N \cup T)^*$ is a regular set. The language

$L(H)$ consists of all words w generated by the underlying grammar G such that there is a derivation tree t of w with respect to G where the words of all levels (except the last one) are in R. The family of all tree controlled grammars is denoted by $\mathcal{TC}$.

Since $R = L(G')$ for some regular grammar $G' = (N', T', P', S')$, the tree controlled grammar H can be given as a pair $H = (G, G')$. As the nonterminal complexity of the tree controlled grammar H, we consider the nonterminal complexity of the underlying context-free grammar G and the nonterminal complexity of the regular grammar G', i.e.,

$$\mathrm{Var}(H) = \mathrm{Var}(G) + \mathrm{Var}(G').$$

For a language L, we set

$$\mathrm{Var}_{\mathcal{TC}}(L) = \min\{\mathrm{Var}(H) : H = (G, G'), \text{ where } G \text{ is a context-free grammar, } G' \text{ is a regular grammar and } L(H) = L\}.$$

An *E0L system* is a quadruple $G = (V, \Sigma, P, \omega)$ where Σ is a non-empty subset of the alphabet V, $\omega \in V^+$ is the axiom, P is a finite subset of $V \times V^*$ which satisfies the condition that, for each $a \in V$, there is a word $w_a \in V^*$ such that $(a, w_a) \in P$ (the elements of P are written as $a \to w_a$). The yield relation $\Rightarrow$ is defined for E0L systems in the following way: $x \Rightarrow y$ holds in G iff the following conditions are satisfied:

(a) $x = a_{i_1} a_{i_2} \cdots a_{i_k}$, $a_{i_j} \in V$ for $1 \le j \le k$,

(b) $y = y_1 y_2 \cdots y_k$, $y_j \in V^*$ for $1 \le j \le k$,

(c) $a_{i_j} \to y_j \in P$ for $1 \le j \le k$.

The language $L(G)$ generated by G is given by

$$L(G) = \{w : w \in \Sigma^*, \omega \Rightarrow^* w\}.$$

An E0L system $G = (V, \Sigma, P, \omega)$ is an *0L system* if $V = \Sigma$, i.e., $G = (\Sigma, P, \omega)$.

An 0L system $G = (\Sigma, P, \omega)$ is *deterministic* if for each $a \in \Sigma$, a rule $(a, w_a) \in P$ is defined uniquely, i.e., for each $a \in \Sigma$, there is only one w_a in Σ^*.

In this paper we use the common name an "*L system*" referring to any of types of defined above.

We denote the language families generated by 0L, D0L and E0L systems by $\mathcal{L}(0L)$, $\mathcal{L}(D0L)$ and $\mathcal{L}(E0L)$, respectively. By definition, the following result follows immediately

Lemma 1.

$$\mathcal{L}(D0L) \subseteq \mathcal{L}(0L) \subseteq \mathcal{L}(E0L).$$

3 Normal Form

In this section, we define a normal form for tree controlled grammars, called *the t-normal form*, and show that for every L system, one can construct equivalent tree controlled grammars in the t-normal form.

Definition 1. *A tree controlled grammar $H = (N, T, P, S, R)$ is said to be in the t-normal form if and only if*

1. $N = N_1 \cup N_2 \cup N_3 \cup \{S\}$ *where* $N_2 = \{A^+ : A \in N_1\}$, $N_3 = \{A^- : A \in N_1\}$, $S \notin N_1 \cup N_2 \cup N_3$;

2. *P may only consist of rules of the following forms*

$$S \to x_1 A_1 S x_2 A_2 S x_3 \cdots x_k A_k S x_{k+1},$$

where $A_i \in N_1$, $1 \le i \le k$, $x_j \in T^*$, $1 \le j \le k+1$,

$$S \to A^- x_1 A_1 S x_2 A_2 S x_3 \cdots x_k A_k S x_{k+1},$$

where $A_i \in N_1$, $1 \le i \le k$, $A^- \in N_3$, $x_j \in T^*$, $1 \le j \le k+1$,

$$S \to A^- x,$$

where $A^- \in N_3$ *and* $x \in T^*$,

$$A \to A^+,\ A^+ \to \varepsilon,\ A^- \to \varepsilon$$

for all $A \in N_1$, $A^+ \in N_2$ *and* $A^- \in N_3$;

3. *the control set is defined by*

$$R = (\{S\} \cup \{A^+ A^- : A \in N_1\} \cup N_1 S \cup T)^*.$$

Before we prove that for every E0L system $G = (V, \Sigma, P, \omega)$, we can construct an equivalent tree controlled grammar in the t-normal form, we slightly modify the definition of the E0L system G by

(1) replacing all terminals a in the rules of P with the new nonterminals $\overline{a}$,

(2) introducing new rules $\overline{a} \to a$ and $a \to a$ for all terminals $a \in \Sigma$,

(3) imposing a restriction on the derivations in G such a way that no terminal appears in any sentential form except the last one.

Definition 2. *An E0L system* $G = (V, \Sigma, P, \omega)$ *is called terminal controlled if*

(a) $\omega \in (V - \Sigma)^*$,

(b) each $(a, w_a) \in P$ *with* $a \in \Sigma$ *has* $w_a = a$,

(c) for every terminal derivation

$$\omega \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n \in \Sigma^*$$

in G, $w_i \in (V - \Sigma)^*$ *for* $1 \le i \le n-1$.

Lemma 2. *For every E0L system* G*, one can construct an equivalent terminal controlled E0L system* G'.

Proof. Let $G = (V, \Sigma, P, \omega)$ be an E0L system. We set $\overline{\Sigma} = \{\overline{a} : a \in \Sigma\}$ and define a homomorphism $\phi : V^* \to ((V - \Sigma) \cup \overline{\Sigma})^*$ by setting $\phi(\varepsilon) = \varepsilon$, $\phi(A) = A$ for all $A \in (V - \Sigma)$, and $\phi(a) = \overline{a}$ for all $a \in \Sigma$.

We construct a terminal controlled E0L system $G' = (V \cup \overline{\Sigma}, \Sigma, P', \overline{\omega})$ where $\overline{\omega} = \phi(\omega)$, and

$$\begin{aligned} P' = &\{A \to \phi(\alpha) : A \to \alpha \in P, A \in (V - \Sigma), \alpha \in V^*\} \\ &\cup \{\overline{a} \to \phi(\alpha) : a \to \alpha \in P, a \in \Sigma, \alpha \in V^*\} \\ &\cup \{\overline{a} \to a : a \in \Sigma\}. \end{aligned}$$

It is not difficult to see that for every terminal derivation

$$\omega \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n \in \Sigma^*$$

in G we can construct a terminal derivation

$$\overline{\omega} \Rightarrow \phi(w_1) \Rightarrow \phi(w_2) \Rightarrow \cdots \Rightarrow \phi(w_n) \Rightarrow w_n$$

in G', where w_n is obtained from $\phi(w_n)$ by applying rules of the form $\overline{a} \to a$, $a \in \Sigma$. On the other hand, for any terminal derivation

$$\overline{\omega} \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_m = w \in \Sigma^*$$

in G', one can construct the terminal derivation

$$\omega \Rightarrow w'_1 \Rightarrow w'_2 \Rightarrow \cdots \Rightarrow w'_{m-1} = w$$

in G, by replacing each nonterminal $\overline{a} \in \overline{\Sigma}$ in $\overline{\omega}$ and w_i, $1 \leq i \leq m-2$ with the terminal $a \in \Sigma$. Thus, $L(G) = L(G')$. □

Remark 1. *Using the same arguments of the proof of Lemma 2, one can easily show that for every (D)0L system there exists an equivalent terminal controlled (D)0L system.*

Lemma 3. *For every L system G, there is an equivalent tree controlled grammar H in the t-normal form.*

Proof. We only prove this lemma for E0L systems, and for other L systems it can be proved analogously.

Let $G' = (V, \Sigma, P, \omega)$ be a terminal controlled E0L system equivalent to an E0L system G. Let $S' \notin V$ is a new symbol. We define two homomorphisms $\varphi : V^* \to (V \cup \{S'\})^*$ and $\varphi' : (V \cup \{S'\})^* \to V^*$ by setting

- $\varphi(\varepsilon) = \varepsilon$, $\varphi(a) = a$ for all $a \in \Sigma$ and $\varphi(A) = AS'$ for all $A \in V - \Sigma$,
- $\varphi'(\varepsilon) = \varepsilon$, $\varphi'(x) = x$ for all $x \in V$ and $\varphi'(S') = \varepsilon$.

We construct a tree controlled grammar $H = (N', T, P', S', R)$ in the t-normal form where $N' = N \cup N^+ \cup N^- \cup \{S'\}$ where

$$N = V - \Sigma, N^+ = \{A^+ : A \in N\}, N^- = \{A^- : A \in N\}, S' \notin N \cup N^+ \cup N^-,$$

and

$$\begin{aligned} P' = &\{S' \to \varphi(\omega)\} \\ &\cup \{S' \to A^-\varphi(\alpha) : A \to \alpha \in P\} \\ &\cup \{A \to A^+ : A \in N\} \\ &\cup \{A \to \varepsilon : A \in N^+ \cup N^-\}, \end{aligned}$$

R is defined as

$$R = (\{S'\} \cup \{A^+A^- : A \in N\} \cup NS')^*.$$

Let $\omega \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n = w \in \Sigma^*$ be a derivation in G'. We construct a terminal derivation D with a derivation tree d in H generating w. Since any sentential form in D – except the last (terminal) one – contains only nonterminals (by the construction of P'), each sentential form in D associated with some level of the derivation tree d is the same as the control word in this level of d.

D starts with $S' \Rightarrow A_1S'A_2S' \cdots A_kS'$ by the rule $S' \rightarrow \varphi(\omega) \in P'$ where $\omega = A_1A_2 \cdots A_k$. If w_1 is obtained by some rules $A_i \rightarrow \alpha_i$, $1 \leq i \leq k$, in D we choose the rules $S' \rightarrow A_i^- \varphi(\alpha_i) \in P'$ and the rules $A_i \rightarrow A_i^+$, which result in

$$D : S' \Rightarrow A_1S'A_2S' \cdots A_kS \Rightarrow^* A_1^+A_1^- \varphi(\alpha_1)A_2^+A_2^- \varphi(\alpha_2) \cdots A_k^+A_k^- \varphi(\alpha_k).$$

If, for all $1 \leq i \leq k$, $\alpha_i \in \Sigma^*$, then

$$w_1 = \alpha_1\alpha_2 \cdots \alpha_k \in \Sigma^*,$$

and $\varphi(\alpha_i) = \alpha_i$, $1 \leq i \leq k$. After erasing the subwords $A_i^+A_i^-$ by $A_i^+ \rightarrow \varepsilon$ and $A_i^- \rightarrow \varepsilon$, the same word w_1 is also obtained in H.

Suppose that for $1 \leq k < n$, $w_k = B_1B_2 \cdots B_m$, $B_i \in N$, $1 \leq i \leq m$, and

$$w' = A_{i_1}^+A_{i_1}^- z_{i_1} A_{i_2}^+A_{i_2}^- z_{i_2} \cdots z_{i_{t-1}} A_{i_t}^+A_{i_t}^- z_{i_t}$$

is the corresponding sentential form in D associated with a level of the derivation tree d. Then, by construction, $z_{i_j} \in (NS')^*$, $1 \leq j \leq t$ and by definition of φ,

$$z_{i_1} z_{i_2} \cdots z_{i_t} = B_1S'B_2S' \cdots B_mS' = \varphi(w_k).$$

Let w_{k+1} is obtained by rules $B_i \rightarrow \beta_i \in P$ for all $1 \leq i \leq m$. Then, in the sentential form w', we choose the rule $B_i \rightarrow B_i^+$ for B_i and the rule $S \rightarrow B_i^- \varphi(\beta_i) \in P'$ for the occurrence of S' following B_i. Moreover, the substrings $A_{i_j}^+A_{i_j}^-$, $1 \leq j \leq t$, are erased by $A^+ \rightarrow \varepsilon$ and $A^- \rightarrow \varepsilon$, which result in the sentential form in D associated with the next level of the derivation tree d:

$$w'' = B_1^+B_1^- \varphi(\beta_1)B_2^+B_2^- \varphi(\beta_2) \cdots B_m^+B_m^- \varphi(\beta_m).$$

If $k = n - 1$, then $\beta_i \in \Sigma^*$, $1 \leq i \leq m$, and $w_n = \beta_1\beta_2 \cdots \beta_m \in \Sigma^*$. Again, $\varphi(\beta_i) = \beta_i$, $1 \leq i \leq m$, and after erasing the subwords $B_i^+B_i^-$, $1 \leq i \leq m$, we obtain the terminal word w_n in H too. Thus $L(G') \subseteq L(H)$.

Let $D : S' \Rightarrow^* w_1 \Rightarrow^* w_2 \cdots \Rightarrow^* w_n = w \in \Sigma^*$ be a derivation in H with a derivation tree d, where each $w_i \in (NS' \cup \{A^+A^- : A \in N\})^*$, $1 \leq i \leq n - 1$, is a sentential form associated with level i of the derivation tree d.

By construction of R, $w_1 = A_1S'A_2S' \cdots A_kS'$ where $\omega = A_1A_2 \cdots A_k$, $A_i \in N$, $1 \leq i \leq k$. Again, by construction R, each A_i, $1 \leq i \leq k$, in w_1 is replaced with A_i^+ (by the rules $A_i \rightarrow A_i^+$) and for the occurrence of S' following A_i, some rule $S' \rightarrow A_i^- \alpha_i \in P'$ is applied, which result in

$$w_2 = A_1^+A_1^- \alpha_1 A_2^+A_2^- \alpha_2 \cdots A_k^+A_k^- \alpha_k.$$

By construction of R, $\alpha_i \in (NS')^*$ for all $1 \leq i \leq k$ or $\alpha_i \in \Sigma^*$ for all $1 \leq i \leq k$.

Case 1. If $\alpha_i \in \Sigma^*$ for all $1 \leq i \leq k$, after erasing the substrings $A_i^+A_i^-$, $1 \leq i \leq k$, the terminal word $w_3 = \alpha_1\alpha_2 \cdots \alpha_k \in \Sigma^*$ is obtained. Then

$$\omega = A_1A_2 \cdots A_k \Rightarrow \alpha_1\alpha_2 \cdots \alpha_k$$

is the derivation in G' simulating D.

Case 2. If $\alpha_i \in (NS')^*$ for all $1 \le i \le k$, then

$$\omega = A_1 A_2 \cdots A_k \Rightarrow \varphi'(\alpha_1)\varphi'(\alpha_2)\cdots\varphi'(\alpha_k)$$

is the derivation in G' simulating D.

Suppose that for $1 < t < n$,

$$w_t = A^+_{i_1} A^-_{i_1} y_1 A^+_{i_2} A^-_{i_2} y_2 \cdots A^+_{i_m} A^-_{i_m} y_m,$$

where $y_j \in (NS')^*$, $1 \le j \le m$, and $A^+_{i_j} \in N^+$, $A^-_{i_j} \in N^-$, $1 \le j \le m$, is a sentential form in D, and $z = \varphi'(y_1)\varphi'(y_2)\cdots\varphi'(y_m)$ is the corresponding sentential form generated in G'.

From w_t, the derivation is continued by erasing the subwords $A^+_{i_j} A^-_{i_j}$ by the rules $A^+_{i_j} \to \varepsilon$ and $A^-_{i_j} \to \varepsilon$, $1 \le j \le m$, by applying for each nonterminal $A \in N$ in y_i, $1 \le i \le m$, the rule $A \to A^+$, and by applying for the occurrence of S' following each A, some rule $S' \to A^-\alpha \in P'$, which result the sentential form $w_{t+1} = y'_1 y'_2 \cdots y'_m$ where $y'_i \in (NS' \cup \{A^+A^- : A \in N\})^*$, $1 \le i \le m$. Then $z' = y''_1 y''_2 \cdots y''_m$ is the corresponding sentential form in G', where y''_i, $1 \le i \le m$, is obtained from y'_i by erasing all occurrences of S' and all subwords of the form A^+A^-.

If $t = n - 1$, then $y_i \in \Sigma^*$ for all $1 \le i \le m$, and $z = y_1 y_2 \cdots y_m = w \in \Sigma^*$. Thus $L(H) \subseteq L(G')$. □

4 A Nonterminal Complexity Bound

In this section we prove that every L language can be generated by a tree controlled grammar with no more than five nonterminals.

Theorem 1. *Every E0L language can be generated by a tree controlled grammar having no more than five nonterminals.*

Proof. Let $L \subseteq \Sigma^*$ be an E0L language generated by the tree controlled grammar $H = (N', T, P', S', R)$ in the t-normal form defined in the proof of Lemma 3. Let $N = \{A_i : 1 \le i \le n\}$ where $S = A_1$, and $A, B, C \notin N'$ be new nonterminals. We define the morphism $\phi : (N \cup N^+ \cup N^-)^* \to \{A, B, C\}^*$ by setting

$$\phi(A_i) = A^i,\ \phi(A^+_i) = \phi(A^-_i) = CB^iC, 1 \le i \le n,$$

and construct the tree controlled grammar $H' = (N'', T, P'', S', R')$ where $N'' = \{S', A, B, C\}$ and P'' consists of the rules

$$S' \to \phi(A_{i_1})S'\phi(A_{i_2})S' \cdots \phi(A_{i_k})S'$$

for the rule of the form

$$S' \to A_{i_1} S' A_{i_2} S' \cdots A_{i_k} S' \in P',$$

$A_{i_j} \in N$, $1 \le j \le k$,

$$S' \to \phi(A^-)\phi(A_{j_1})S'\phi(A_{j_2})S' \cdots \phi(A_{j_l})S'$$

for each rule of the form

$$S' \to A^- A_{j_1} S' A_{j_2} S' \cdots A_{j_l} S' \in P',$$

$A \in N$, $A_{j_i} \in N$, $1 \le i \le l$,

$$S' \to \phi(A^-)x$$

for each rule of the form

$$S' \to A^- x \in P',$$

$A \in N$, $x \in \Sigma \cup \{\varepsilon\}$, and the chain as well as erasing rules

$$A \to B,\ B \to \varepsilon,\ C \to \varepsilon.$$

The control set R' is defined as

$$R' = (\{S'\} \cup \{A^i S' : 1 \le i \le n\} \cup \{B^i C B^i C : 1 \le i \le n\})^*.$$

Any derivation D in the grammar H can be directly simulated by the derivation D' in the grammar H' as follows:

- the application of

  $$S' \to A_{i_1} S' A_{i_2} S' \cdots A_{i_k} S'$$

  is replaced by

  $$S' \to \phi(A_{i_1}) S' \phi(A_{i_2}) S' \cdots \phi(A_{i_k}) S';$$

- the application of a rule

  $$S' \to A^- A_{j_1} S' A_{j_2} S' \cdots A_{j_l} S'$$

  or

  $$S' \to A^- x,$$

  $x \in T \cup \{\varepsilon\}$ is replaced by

  $$S' \to \phi(A^-) \phi(A_{j_1}) S' \phi(A_{j_2}) S' \cdots \phi(A_{j_l}) S'$$

  or

  $$S' \to \phi(A^-) x,$$

  respectively;

- the application of $A_i \to A_i^+$, $1 \le i \le n$, is replaced by the application of the sequence of the rules

  $$\underbrace{A \to B, \ldots, A \to B}_{i \text{ times}};$$

- the applications of $A_i^+ \to \varepsilon, A_i^- \to \varepsilon$, $1 \le i \le n$, are replaced by the applications of the sequences of the rules

  $$\underbrace{B \to \varepsilon, \ldots, B \to \varepsilon}_{i \text{ times}}$$

  and

  $$C \to \varepsilon, \underbrace{B \to \varepsilon, \ldots, B \to \varepsilon}_{i \text{ times}}, C \to \varepsilon,$$

respectively, and it is not difficult to see that the words at the levels of the derivation tree for D' is in R'. Thus, $L(H) \subseteq L(H')$.

Let

$$D' : S' \Rightarrow w_1 = A^{i_1} S' A^{i_2} S' \cdots A^{i_k} S' \Rightarrow^* w_n \in \Sigma^*$$

be a terminal derivation in H' with a derivation tree d, where each

$$w_i \in (\{S'\} \cup \{A^i S' : 1 \leq i \leq n\} \cup \{B^i C B^i C : 1 \leq i \leq n\})^*, 1 \leq i \leq n,$$

is a sentential form associated with level i of the derivation tree d.

It is not difficult to see that w_2 is obtained from w_1 by applying $A \to B$ for all occurrences of A in w_1 and applying for each S' following A^{i_j}, $1 \leq j \leq i_k$, some rule of the form $S' \to CB^{i_j}C\alpha_j$ where

$$\alpha_j \in (\{A^i S' : 1 \leq i \leq n\})^* \text{ or } \alpha_j \in \Sigma^*,$$

i.e.,

$$w_2 = B^{i_1}CB^{i_1}C\alpha_{i_1}B^{i_2}CB^{i_2}C\alpha_{i_2} \cdots B^{i_k}CB^{i_k}C\alpha_{i_k}.$$

Then the derivation D in the grammar H simulates D' as follows:

$$S' \Rightarrow w'_1 = A_{i_1}S'A_{i_2}S' \cdots A_{i_k}S' \Rightarrow^* A^+_{i_1}A^-_{i_1}\alpha'_{i_1}A^+_{i_2}A^-_{i_2}\alpha'_{i_2} \cdots A^+_{i_k}A^-_{i_k}\alpha'_{i_k}.$$

where $\alpha'_{i_j} = \alpha_{i_j}$, $1 \leq j \leq k$, if $\alpha_{i_j} \in \Sigma^*$ or it is obtained from α_{i_j} by replacing each substring of the form A^l by nonterminal A_l.

$$w_t \in (\{A^i S' : 1 \leq i \leq n\} \cup \{B^i C B^i C : 1 \leq i \leq n\})^*,$$

and it results in w_{t+1} by (1) erasing all occurrences of B, C, (2) replacing all occurrences of A with B, and (3) applying some $S' \to CB^iC\alpha \in P''$ for the occurrence of S' following the substring A^i.

Correspondingly, w'_{t+1} in D is obtained from w'_t by (1') erasing all occurrences of all nonterminals of the form A^+, A^-, (2') replacing each A_j in w'_t with A^+_j, and (3') applying some $S' \to A^-_j\alpha' \in P'$ for the occurrence of S' following A_j, where α' is obtained from α_{i_j} by replacing each substring of the form A^l by nonterminal A_l.

If $t = n-1$, then each α in (3) is a terminal string, and $\alpha' = \alpha$ in (3'). Since the terminal string w_n in D' is obtained by erasing all occurrences of B, C in w_{n-1}, we similarly erase all occurrences of all nonterminals of the form A^+, A^- in w'_{n-1}, and obtain the same string w_n in H''.

Thus $L(H') \subseteq L(H)$. Since R' can be generated by a regular grammar with one nonterminal symbol, $\text{Var}(H') = 5$ and $\text{Var}_{\mathcal{TC}}(L) \leq 5$. □

From Lemma 1 and Theorem 1 the following result follows immediately

Corollary 1. *Every (D)0L language can be generated by a tree controlled grammar with no more than five nonterminals.*

5 Conclusions

In this paper we have established a bound five for nonterminal complexity for tree controlled grammars generating L languages, using the t-normal form for tree-controlled grammars with one "active" nonterminal and a coding homomorphism. But it remains open if this bound is optimal. We also do not know a good bound for T0L and ET0L languages.

Acknowledgments

This work has been supported by Ministry of Higher Education Fundamental Research Grant Scheme FRGS /1/11/SG/UPM/01/1 and University Putra Malaysia via RUGS 05-01-10-0896RU/F1.

References

[1] N. Chomsky. Three models for the description of languages. *IRE Trans. on Information Theory*, 2(3):113–124, 1956.

[2] N. Chomsky. *Syntactic structure.* Mouton, Gravenhage, 1957.

[3] N. Chomsky. On certain formal properties of grammars. *Information and Control*, 2:137–167, 1959.

[4] J. Dassow and Gh. Păun. *Regulated rewriting in formal language theory.* Springer-Verlag, Berlin, 1989.

[5] K. II Culik and H. Maurer. Tree controlled grammars. *Computing*, 19:129–139, 1977.

[6] A.B. Cremers, O. Mayer, and K. Weiss. On the complexity of regulated rewriting. *Information and Control*, 33:10–19, 1974.

[7] J. Dassow. Remarks on the complexity of regulated rewriting. *Fundamenta Informaticae*, 7:83–103, 1984.

[8] G. Păun. Six nonterminals are enough for generating each RE language by a matrix grammar. *International Journal of Computional Mathematics*, 15:23–37, 1984.

[9] J. Dassow and G. Păun. Further remarks on the complexity of regulated rewriting. *Kybernetika*, 21:213–227, 1985.

[10] J. Dassow and G. Păun. Some notes on the complexity of regulated rewriting. *Bull. Math. Soc. Sci. Math. R.S. Roumanie*, 30:203–212, 1986.

[11] H. Fernau. Nonterminal complexity of programmed grammars. *Theoretical Computer Science*, 296:225–251, 2003.

[12] E. Csuhaj-Varjú and G. Vaszil. Scattered context grammars generated any recursively enumerable language with two nonterminals. *Information Processing Letters*, 1106:902–907, 2010.

[13] H. Fernau and A. Meduna. On the degree of scattered context-sensitivity. *Theoretical Computer Science*, 290:2121–2124, 2003.

[14] H. Fernau and A. Meduna. A simultaneous reduction of several measures of descriptional complexity in scattered context grammars. *Information Processing Letters*, 86:235–240, 2003.

[15] G. Vaszil. On the descriptional complexity of some rewriting mechanisms regulated by context conditions. *Theoretical Computer Science*, 330:361–373, 2005.

[16] A. Meduna and A. Gopalaratnam. On semi-conditional grammars with productions having either forbidding or permitting conditions. *Acta Cybernetica*, 11:307–323, 1994.

[17] A. Meduna and M. Švec. Reduction of simple semi-conditional grammars with respect to the number of conditional productions. *Acta Cybernetica*, 15:353–360, 2002.

[18] S. Turaev, J. Dassow, and M. Selamat. Nonterminal complexity of tree controlled grammars. *Theoretical Computer Science*, 412:5789–5795, 2011.

[19] S. Turaev, J. Dassow, and M. Selamat. Language classes generated by tree controlled grammars with bounded nonterminal complexity. In M. Holzer, M. Kutrib, and G. Pighizzini, editors, *Descriptional Complexity of Formal Systems*, volume 6808 of *LNCS*, pages 289–300. Springer, Heidelberg, Berlin, 2011.

[20] S. Turaev, J. Dassow, F. Manea, and M. Selamat. Language classes generated by tree controlled grammars with bounded nonterminal complexity. *Theoretical Computer Science*, 449:134–144, 2012.

[21] G. Rozenberg and A. Salomaa, editors. *Handbook of formal languages*, volume 1–3. Springer-Verlag, 1997.