Provided for non-commercial research and educational use.
Not for reproduction, distribution or commercial use.

Mathematica Balkanica

Mathematical Society of South-Eastern Europe
A quarterly published by
the Bulgarian Academy of Sciences – National Committee for Mathematics

---

The attached copy is furnished for non-commercial research and education use only. Authors are permitted to post this version of the article to their personal websites or institutional repositories and to share with other researchers in the form of electronic reprints.
Other uses, including reproduction and distribution, or selling or licensing copies, or posting to third party websites are prohibited.

For further information on Mathematica Balkanica visit the website of the journal
http://www.mathbalkanica.info
or contact:
Mathematica Balkanica - Editorial Office;
Acad. G. Bonchev str., Bl. 25A, 1113 Sofia, Bulgaria
Phone: +359-2-979-6311, Fax: +359-2-870-7273,
E-mail: balmat@bas.bg

Mathematica Balkanica

New Series Vol. 5, 1991, Fasc. 1

On Lindenmayer Systems and String Adjunct Schemes

Nguen Xuan My

Presented by P. Kenderov

1. Introduction

The theory of L systems were originated by Aristid Lindenmayer in connection with biological considerations (see [4]). The basic model of Lindenmayer systems is a device generating languages by means of several finite substitutions. There were many attempts to generalize this model where instead of finite substitutions, one used more general mappings (see [14], [13], [15], [10], [5], [1], for details).

In [8], we have introduced and studied generalized Lindenmayer systems which are defined similarly to the Lindenmayer's ones but instead of finite substitutions, we used the pairs consisting of a finite substitution and a domain on which the substitution may activate. This model is a special type of mapping grammars proposed in [5]. In [8], we considered generalized systems in which the active domains of finite substitutions are regular languages.

In this paper, we shall study generalized Lindenmayer systems with the active domains of finite substitutions belonging to different families of the Chomsky hierarchy. We shall find out a describing of string adjunct schemes (see [3] or [7] for this notion) by means of generalized Lindenmayer systems. This result claims that the notion of generalized Lindenmayer system would be an appropriate framework for unifying the both theories of Lindenmayer systems and of string adjunct schemes. On the other hand, by virtue of this interrelation, we can prove some results concerning generalized Lindenmayer systems.

The paper will be organized as follows. Section 2 will be devoted to preliminaries for terminologies, necessary definitions and well-known results. In Section 3, we shall formulate and prove the results.

2. Preliminaries

We assume that the reader is familiar with the notions, notations and main results of formal language theory which can be found in, e. g. [9] or [12]. We shall specify here some of them. Every finite set of symbols is called an alphabet. Let

V be an alphabet, we denote by V^* the free monoid generated by V with the unity denoted by λ, and called the empty string. Every element of V^* is said a string over V. For a string x, $|x|$ denotes its length. Every subset of V^* is called a language over V. V^* is termed the whole language.

The Chomsky hierarchy is the properly increasing sequence of families of regular, context-free, context-sensitive, recursive and recursively enumerable, respectively, languages, and exprimed under the form

$$£(\mathrm{REG}) \subset £(\mathrm{CF}) \subset £(\mathrm{CS}) \subset £(\mathrm{REC}) \subset £(\mathrm{RE}).$$

We shall also use the conventional inequalities:

$$\mathrm{REG} < \mathrm{CF} < \mathrm{CS} < \mathrm{REC} < \mathrm{RE}$$

correspondent to the above inclusions.

As regards to the Lindenmayer system theory, we refer to [11].

Definition 2.1. A generalized Lindenmayer system (a GLS for short) is one of the form

$$M = (V, \{(F_i, f_i), 1 \leqq i \leqq k\}, w)$$

where for every i, $1 \leqq i \leqq k$, F_i is a language over V, $f_i: V^* \to V^*$ is a finite substitution, and w is a string over V.

For two arbitrary strings x and y over V, we write $x \Rightarrow y$ or say x derives y if there exists an i, $1 \leqq i \leqq k$, such that x is in F_i, and y is in $f_i(x)$ (Note that in general, f_i's are set-valued).

The language generated by M is defined by the set

$$L(M) = \{x \text{ in } V^* | w \Rightarrow^* x\}$$

where $\Rightarrow^*$ denotes the reflexive and transitive closure of $\Rightarrow$.

Definition 2.2. An extended generalized Lindenmayer system (an EGLS for short) is one of the form

$$G = (V, \{(F_i, f_i), 1 \leqq i \leqq k\}, w, T)$$

where $M_G = (V, \{(F_i, f_i), 1 \leqq i \leqq k\}, w)$ is a GLS, and T is a subset of V, called the set of terminals.

The language generated by G, denoted by $L(G)$, is defined by

$$L(G) = L(M_G) \cap T^*.$$

Remark 2.3. We can enumerate here some special types of GLS's and EGLS's:

(i) A GLS in which $k=1$ and $F_1 = V^*$ is exactly an OL system (cf. [11]).
(ii) A GLS in which all F_i's are V^* is exactly a TOL system (cf. [11]).
(iii) An EGLS in which $k=1$ and $F_1 = V^*$ is exactly an EOL system (cf. [11]).
(iv) An EGLS in which all F_i's are V^* is exactly an ETOL system (cf. [11]).

Definition 2.4. For X in {OL, TOL, EOL, ETOL}, and for Y in {REG, CF, CS, REC, RE}, an $Y-X$ system is defined as an X system but instead of F_i's are V^*, it is requested that F_i's belong to £(Y).

Remark 2.5. In [8], one introduced and studied REG-X systems for X in {OL, TOL, EOL, ETOL}, under the name "generalized X system".

Convention 2.6. For a language generating device of type P, every language generated by a P device will be said a P language and £(P) will denote the family of P languages.

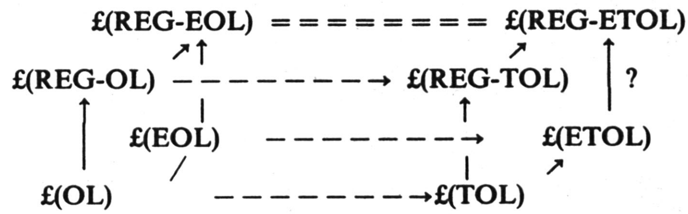
Let us now remember some results related to REG-X systems for X in {OL, TOL, EOL, ETOL}.

Proposition 2.7. ([8]) (*i*) £(REG-OL) *is not closed under union, catenation, Kleene closures, intersection with regular sets, nonerasing homomorphisms and inverse homomorphisms.*

(*ii*) £(REG-TOL) *has the same nonclosure properties as that of* £(REG-OL), *but its behaviour under Kleene closures is not known yet.*

(*iii*) *The both families* £(REG-EOL) *and* £(REG-ETOL) *are closed under all the six above operations. Moreover, they are closed under finite substitutions and are hyper*-AFL's (*see* [13] *for this notion*).

Proposition 2.8. ([8]) *The interrelations between the families* £(X)*'s and* £(REG-X) *for* X *in* {OL, TOL, EOL, ETOL} *are described by the following diagram in which the arrow stands for proper inclusion and* $\overset{?}{- - - - \rightarrow}$ *stands for the inclusion which is not known yet proper or not. Note that for Lindenmayer systems, we have the strict inclusion* £(EOL)⊂£(ETOL), *whereas* £(REG-EOL)= £(REG-ETOL).



Definition 2.9. Let M be a GLS according to Definition 2.1. If $x \Rightarrow y$ by the pair (F_i, f_i) then we also write $x \overset{i}{\Rightarrow} y$. We define the notion of control word of a derivation as follows: for every string x over V, the control word of $x \Rightarrow^* x$ is λ; for a derivation

$$D: x_0 \overset{i_0}{\Rightarrow} x_1 \overset{i_1}{\Rightarrow} x_2 \overset{i_2}{\Rightarrow} x_3 \ldots x_{n-1} \overset{i_{n-1}}{\Longrightarrow} x_n$$

we call the string $i_0 i_1 i_2 \ldots i_{n-1}$ over the alphabet $\{1, 2, \ldots, k\}$ a control word of D, denoted by contr (D).

Let C be a language over the alphabet $\{1, 2, \ldots, k\}$ and M(or G) be a GLS (resp. an EGLS). The language generated by M(resp. by G) with the control language C is defined by the set

$$L(M, C)=\{x \text{ in } V^* \mid \exists D: w \overset{*}{\Rightarrow} x, \text{ contr } (D) \text{ in } C\}$$

$$(\text{resp. } L(G, C)=\{x \text{ in } T^* \mid \exists D: w \overset{*}{\Rightarrow} x, \text{ contr } (D) \text{ in } C\}).$$

For X in {OL, TOL, EOL, ETOL} and Y, Z in {REG, CF, CS, REC, RE}, we denote by $\pounds(Y-X, Z)$ the family of languages generated by $Y-X$ systems with control languages belonging to $\pounds(Z)$.

Proposition 2.10. *For X in* {EOL, ETOL} *and Y in* {REG, CF, CS, REC, RE}, $\pounds(Y-X, Y)=\pounds(Y-X)$.

Proof. Since all the families $\pounds(Y)$ are closed under catenation, for this result, we can repeat the proof for the equality £(REG-ETOL, REG)=£(REG-ETOL) in [8].

Definition 2.11. (see [3] or [7]). A string adjunct scheme of degree n (briefly, an n-SAS) is a system of the form

$$S=(V, \$, \{(F_t, u_t), 1 \leq t \leq k\}, B)$$

where V and $\$$ are disjoint alphabets, V is the set of terminals and $\$=\{\$_1, \$_2, \ldots, \$_n\}$ is the set of markers, B is a finite language over V, and for every t, $1 \leq t \leq k$, F_t is a subset of the language $V^*\$_1V^*\$_2 \ldots V^*\$_nV^*$ and u_t is an n-tuple of strings over V of the form $u_t=(u_{t1}, u_{t2}, \ldots, u_{tn})$.

An n-SAS is λ-free if for every t, $1 \leq t \leq k$, the string $u_{t1} u_{t2} \ldots u_{tn}$ is nonempty.

The derivation of strings according to S is defined as follows: for x, y in V^*, $x \Rightarrow y$ if there exists t, $1 \leq t \leq k$, $x_1 \$_1 \ldots x_n \$_n x_{n+1}$ in F_t, such that $x=x_1 \ldots x_n x_{n+1}$ and $y=x_1 u_{t1} x_2 u_{t2} \ldots x_n u_{tn} x_{n+1}$.

The language generated by S, denoted by $L(S)$, is defined by the set

$$L(S)=\{x \text{ in } V^* \mid b \Rightarrow^* x \text{ for some } b \text{ in } B\}.$$

Similarily as for GLS's, for n-SAS we can define the notion of languages generated by n-SAS's with control languages.

An n-SAS is called (n, Y)-SAS for Y in {REG, CF, CS, REC, RE}, if all F_t's are in $\pounds(Y)$.

We denote by $\pounds(n, Y, Z)$ the family of languages generated by (n, Y)-SAS's with control languages belonging to $\pounds(Z)$ for Y, Z in {REG, CF, CS, REC, RE}, and $\pounds(n\text{-}\lambda, Y, Z)$ the family of languages generated by λ-free (n, Y)-SAS's with control languages belonging to $\pounds(Z)$.

Proposition 2.12. (see [6]). (*i*) *For every* $n \geq 1$,

$$\pounds(n, Y, Z)=\pounds(\text{RE}) \text{ for all } Y \text{ and } \text{CS} \leq Z \leq \text{RE}$$

$$\pounds(n, Y, Z)=\pounds(Y) \text{ for } Z=\text{REG, CF and } \text{CS} \leq Y \leq \text{RE}$$

and for REG$\leqq Y, Z \leqq$CF, $(Y, Z) \neq$(REG, REG), $\mathcal{L}(n, Y, Z)$ *properly lies between* $\mathcal{L}$(CF) *and* $\mathcal{L}$(CS), *and* $\mathcal{L}(n$, REG, REG)—*between* $\mathcal{L}$(REG) *and* $\mathcal{L}$(CS).

(*ii*) *For every* $n \geqq 1$,

$\mathcal{L}(n-\lambda, Y, Z) = \mathcal{L}(\max\{Y, Z\})$ *for* REG$\leqq Y \leqq$RE *and* CS$\leqq Z \leqq$RE *or for* REG$\leqq Z \leqq$RE *and* CS$\leqq Y \leqq$RE.

$$\mathcal{L}(n-\lambda, Y, Z) = \mathcal{L}(n, Y, Z) \text{ for } \mathrm{REG} \leqq Y,\ Z \leqq \mathrm{CF}.$$

(*iii*) *All the families* $\mathcal{L}(n, Y, Z)$ *and* $\mathcal{L}(n-\lambda, Y, Z)$ *are closed under intersection with regular languages.*

3. Some results

In this section, we shall prove some results on n-SAS's and generalized Lindenmayer systems from which the two following are remarkable. One claims that every n-SAS is equivalent to an EGLS, this means that the theory of generalized Lindenmayer is enough to unify the both of L systems and of string adjunct schemes. Another shows that the family of context-free languages is strictly included in the family of languages generated by (1, REG)-string adjunct schemes with control languages belonging to $\mathcal{L}$(REG). This means that for generating every context-free language, it is sufficient to use a two-fold regular mechanism for adding appropriate strings to strings of a finite language: one for choosing which strings are added, and another for regulating the adding process.

Proposition 3.1. (*i*) *For every* n-SAS

$$S = (V, \$, \{(F_t, u_t), 1 \leqq t \leqq k\}, B)$$

there exists an EGLS G *such that* $L(S) = L(G)$. *Moreover, by the constructing of* G, *for* X=REG, CF, CS, REC, RE, *if* S *is an* (n, X)-SAS *then* G *is an* X-ETOL *system.*

(*ii*) *If* C *is a control language for* S *then there exists a control language* C' *for* G, *which is obtained from* C *under a nonerasing homomorphism such that* $L(S, C) = L(G, C')$.

Proof. (i) G is constructed as follows:

$$G = (V \cup \$ \cup \{Z\}, \{(F_t, f_t), 1 \leqq t \leqq k, (V^*, f)\}, Z, V)$$

where Z is a new symbol belonging not to $V \cup \$$, and for every t, $1 \leqq t \leqq k$, f_t is a finite substitution defined by

$$f_t(a) = a \text{ for } a \text{ in } V$$

$$f_t(\$_i) = u_{ti} \text{ for } 1 \leqq i \leqq n$$

$$f_t(Z) = Z$$

and f is a finite substitution defined by

$$f(a)=\{a,\ a\$_i,\ \$_i a,\ 1\leqq i\leqq n\}$$
$$f(\$_i)=\$_i \text{ for } 1\leqq i\leqq n$$
$$f(Z)=B$$

First of all, by means of f, G can generate B and from B-strings again by f, G derivates strings from which the strings belonging to F_t for some t may be added by u_{ti}'s by means of f_t.

According to this explanation, we can verify that $L(G)=L(S)$.

Obviously, if S is (n, X)-SAS then G is X-ETOL system for $X=$ REG, ..., RE.

(ii) Let C be a control language for S, i.e. C is a language over $\{1, 2, \dots, k\}$. Let C' be the language over $\{f, f_t, 1\leqq t\leqq k\}$ defined by $C'=f.\ h(C)$ where h is the nonerasing homomorphism defined by $h(i)=ff_i$, $1\leqq i\leqq k$. Again by virtue of the same above explanation, we can verify that the language generated by S with the control language C equals to that generated by G with the control language C'.

From this proposition and by taking into account that all the families £(X) for $X=$ REG, ..., RE, are closed under nonerasing homomorphisms, we obtain the following result

Corollary 3.2. *For every* $n\geqq 1$, Y, Z *in* REG, CF, CS, REC, RE, *the family* £(n, Y, Z) *is contained in the family* £(Y-ETOL, Z) *and* £(n, Y) *– in* £(Y-ETOL).

By means of this corollary, Propositions 2.10, 2.12 and the Church's Thesis, we have the following result dealing with the interrelation between the families £(Y-ETOL, Z) and the Chomsky hierarchy.

Proposition 3.3. £(Y-ETOL, Z)=£(RE) *for* REG$\leqq Y\leqq$RE *and* CS$\leqq Z\leqq$RE *or for* CS$\leqq Y\leqq$RE *and* REG$\leqq Z\leqq$RE.

Proposition 3.4. *The family* £(CF) *is properly included in the family* £(1, REG, REG).

Proof. The idea of the proof is as follows. By virtue of the Schutzenberger's theorem, for every context-free language L, there exist a regular language R, a homomorphism h such that $L=h(D\cap R)$ where D is the Dyck language. Therefore, for proving the inclusion £(CF)$\subseteq$£(1, REG, REG), it is sufficient to show that:

(i) D belong to £(1, REG, REG).

(ii) £(1, REG, REG) is closed under intersection with regular languages.

(iii) £(1, REG, REG) is closed under homomorphisms.

According to (iii) of Proposition 2.12, (ii) holds true.

For (iii), let

$$S=(V,\ \$,\ \{(F_t,\ u_t),\ 1\leqq t\leqq k\},\ B)$$

be an n-SAS with a control language C over $\{1, 2, \dots, k\}$ where F_t's and C are regular, and h be a homomorphism. Then we can verify that $h(L(S, C))$ is generated by the n-SAS

$$S_h=(V,\ \$,\ \{(F_t^h,\ u_t^h),\ 1\leqq t\leqq k\},\ h(B))$$

with the same control language C where F_t^h's and u_t^h's are obtained from F_t's and u_t's, respectively, under the homomorphism $a \rightsquigarrow h(a)$ for a in V, and $\$_i \rightsquigarrow \$_i$ for $1\leqq i\leqq n$. Obviously, all F_t^h's are still regular, therefore £(n, REG, REG) is closed under homomorphisms for $n\geqq 1$.

For (i), let D be generated by the context-free grammar

$$G=(V=\{A,\ x_1,\dots,x_{2m}\},\ P,\ A,\ \{x_1,\dots,x_{2m}\})$$

where P is the set of rewriting rules of the form

1. $A \dashrightarrow x_i A x_{i+m} A$
2. $A \dashrightarrow x_{i+m} A x_i A \quad 1\leqq i\leqq m.$
3. $A \dashrightarrow \lambda$

We denote by SENT(G) the set of all sentential forms of G obtained by applying only the rules of the form 1 and 2 in left-most derivations.

Consider the following 1-SAS

$$G=(V,\ \$,\ \{(F_r,\ u_r),\ 1\leqq r\leqq 4m\},\ \{A\})$$

where for $1\leqq r\leqq m$,

$$F_r=\{x_1,\dots,x_{2m}\}^*\ \$AV^*, \qquad u_r=x_r$$
$$F_{m+r}=F_r, \qquad u_{m+r}=x_{m+r}$$
$$F_{2m+r}=\{x_1,\dots,x_{2m}\}^*\ A\$V^*, \qquad u_{2m+r}=x_{m+r}$$
$$F_{3m+r}=F_{2m+r}, \qquad u_{3m+r}=x_r.$$

We can verify that SENT(G) is generated by G with the control language $\{r.(2m+r),\ (m+r).(3m+r),\ 1\leqq r\leqq m\}^*$ hence SENT(G) belongs to £(1, REG, REG). On the other hand, D is obtained from SENT(G) under the homomorphism $x_i \rightsquigarrow x_i$ for $1\leqq i\leqq 2m$ and $A \rightsquigarrow \lambda$ therefore D belongs to £(1, REG, REG) too.

According to (i)-(iii), the inclusion £(CF)⊂£(1, REG, REG) holds true.

For its strictness, we note that the language

$$L=\{a^n b^n a^n \mid n\geqq 1\}$$

is not context-free, but L is generated by the 1-SAS

$$(\{a,\ b\},\ \$,\ \{(a^+\ \$b^+\ a^+,\ ab),\ (a^+\ b^+\ \$a^+,\ a)\},\ \{aba\})$$

with the control language $(1.2)^*$ where 1 and 2 stand for the pairs $(a^+\ \$b^+\ a^+,\ ab)$ and $(a^+\ b^+\ \$a^+,\ a)$, respectively.

The result of Proposition 3.4 is an affirmative answer for an open problem formulated in [6].

Note that £(1, REG, REG)⊆£(REG-ETOL)=£(REG-EOL) and in general £(n, REG, REG)⊆£(REG-ETOL)=£(REG-EOL). On the other hand, £(EOL) is also strictly included in £(REG-EOL). The following result will give a relation between £(1, REG, REG) and £(EOL).

Proposition 3.5. £(1, REG, REG)—£(EOL)$\neq\emptyset$.

Proof. It is known that the language

$$L=\{a^k b^l a^k \mid 1\geqq k\geqq 1\}$$

does not belong to £(EOL) (see [11]). However L is generated by the (1, REG)-SAS

$$(\{a,\ b\},\ \$,\ \{(a^+\$b^+a^+,\ ab),\ (a^+b^+\$a^+,\ a),\ (a^+\$b^+a^+,\ b)\},\ \{aba\})$$

with the control language $(1.2)^+3^*$ where 1, 2, and 3 stand for the pairs $(a^+\$b^+a^+, ab)$, $(a^+b^+\$a^+, a)$ and $(a^+\$b^+a^+, b)$, respectively.

Proposition 3.6. *The family* £(ETOL) *is properly contained in the family* £(CF-ETOL).

Proof. According to Proposition 2.10, £(CF-ETOL)=£(CF-ETOL, CF). In view of [0], the language

$$L=\{(a^{m+1}b)^{n-1}a^{m+1}c \mid m\geqq 0,\ n=2^{m+1}\}$$

does not belong to £(ETOL) but it is generated by a TOL system with context-free control language, therefore L belongs to £(CF-ETOL, CF)—£(ETOL), that claims the strictness of the obvious inclusion £(ETOL)$\subset$£(CF-ETOL).

Remark 3.7. We do not know if the inclusion £(ETOL)$\subseteq$£(REG-ETOL) is strict. If there happens the equality £(CS)=£(REG-EOL, CF), then the family £(REG-ETOL) is properly included in £(CS). We conjecture that £(CS)=£(REG-EOL, CF).

Acknowledgement. The author would like to thank the leadership of Institute of Mathematics with Computing centre, Bulgarian Academy of Sciences and his colleagues in Department of Mathematical Linguistics for their helping and encouragement, and would like to dedicate to them this paper.

References

00. P. R. J. Asveld. Control iteration grammars and full hyper-AFL's. **Inform. Contr., 34**, 1977, 248-269.
01. P. R. J. Asveld. Abstract grammars based on transactions, *Technische Hogeschool Twente, Memorandum INF-85-13*, 1985.
02. J. Berstel. Transactions and context-free languages. B. G. Teubner, Stuttgart, 1979.
03. S. Israil, Nguyen Xuan My. N-contextual scheme. *Bull. Math. Soc. Sci. RSR*, **26**, 1982, 241-249.
04. A. Lindenmayer. Mathematical models for cellular interactions in development (I+II). *J. Theor. Biology*, **18**, 1968, 280-315.
05. Nguyen Xuan My. A structural model for systems. *Proc. of the Annual Scientific Conference*, Inst. Math., Hanoi, 1983.
06. Nguyen Xuan My. String adjunct schemata with control languages. *Rev. Roum. Math. pures et appl.*, **28**, 1983, 715-730.
07. Nguyen Xuan My. String adjunct schemes.—In: Proc. of the 19th Spring Conf. of the Union of Bulgarian Mathematicians, 1990.

08. Nguyen Xuan My. Generalized Lindenmayer systems. Submitted to *Theoretical Computer Science*, 1990.
09. Р. Д. Павлов. Математическа лингвистика: увод в теорията на формалните езици и граматики. София, 1982 (in Bulgarian).
10. B. Rovan. A framework for studying grammars. *Lect. Notes in Computer Science*, **118**, 1981, 473-482.
11. G. Rozenberg, A. Salomaa. The mathematical theory of *L* systems. Acad. Press, New York, 1980.
12. A. Salomaa. Formal languages. Acad. Press, New York, 1973.
13. A. Salomaa. Macros, Iterated substitution and Lindenmayer's AFL's. *DAIMI PB-18*, *Univ. Aarhus*, Aarhus, Denmark, 1973.
14. J. van Leeuwen. F-iteration languages. *Memorandum Univ. California*, Berkley, 1973.
15. D. Wood. Iterated a-NGSM maps and г-system. *Inform. Contr.*, **32**, 1976, 1-26.

Received 15. 03. 1990

Dept. Mathematical Linguistics
Institute of Mathematics with Computing Centre
Bulgarian Academy of Sciences
Bl. 8 — Sofia 1113
BULGARIA