Some Recent Developments in Formal Language Theory

G. Rozenberg

Given a finite alphabet $\Sigma$, any subset of $\Sigma^*$ (the set of all words over $\Sigma$) is called a language. This is how language is defined in formal language theory. Hence in mathematical terms, formal language theory investigates subsets of free monoids. In traditional formal language theory (initiated by the linguist Chomsky) languages are defined by various finitary processes such as grammars and automata, which in general are rather involved combinatorial objects. However in recent years one witnesses a vigorous development of a new area—theory of $L$ systems (initiated by the biologist A. Lindenmayer)—where language definitions have very natural algebraic character.

The aim of this note is to survey a fragment of the theory of $L$ systems. Although we survey only a small portion of available results, it is chosen (and arranged) in such a way as to give the reader some picture of how the theory is built up. We start with the most rudimental systems (consisting of iterations of single homomorphisms) and then consider their three most natural generalizations (namely iterations of a finite number of homomorphisms, iterations of finite substitutions and iterations of a finite number of finite substitutions). In the first section we present several results and research areas typical for this part of the theory which considers sequences of words rather than their sets (languages). This is a subject matter very novel in formal language theory. In the remaining three sections we consider various types of $L$ languages. We concentrate on results describing their combinatorial structure.

We do not quote here the origins of listed results as they all appear in [1] or [2] where they are properly referenced. We use mostly standard terminology and notation. For a word $x$ we use $|x|$ to denote its length and $\#_B x$ to denote the number
of occurrences of letters from \( B \) in \( x \), \( A \) denotes the empty word. For a language \( K \), \( \pi_n(K) \) denotes the number of different subwords of length \( n \) occurring in words from \( K \) and \( \text{Length}(K) = \{ n : n = |x| \text{ for some } x \in K \} \). Given an alphabet \( \Sigma \) and \( \Delta \subseteq \Sigma \), \( \text{Pres}_\Delta \) is a homomorphism from \( \Sigma^* \) into \( \Sigma^* \) defined by \( \text{Pres}_\Delta(a) = a \) for \( a \in \Delta \) and \( \text{Pres}_\Delta(a) = A \) for \( a \in \Sigma \setminus \Delta \).

I. Single homomorphisms iterated. The simplest way to define a sequence of words or a language in \( L \) systems theory is to iterate a homomorphism on a free monoid. It is formalized as follows. A DOL system is a triple \( G = (\Sigma, h, \omega) \) where \( \Sigma \) is a finite alphabet, \( h : \Sigma^* \to \Sigma^* \) is a homomorphism and \( \omega \in \Sigma^+ \). The sequence of \( G \), denoted \( E(G) \), is defined by \( E(G) = \omega_0, \omega_1, \ldots \) where \( \omega_0 = \omega \) and \( \omega_{i+1} = h(\omega_i) \) for \( i \geq 0 \). The language of \( G \), denoted \( L(G) \), is defined by \( L(G) = \{ h^n(\omega) : n \geq 0 \} \). The length sequence of \( G \), denoted \( LS(G) \), is defined by \( LS(G) = |\omega_0|, |\omega_1|, |\omega_2|, \ldots \). The growth function of \( G \), denoted \( f_G \), is the function from nonnegative integers into nonnegative integers defined by \( f_G(n) = |h^n(\omega_0)|, n \geq 0 \). \( E(G) \) is referred to as a DOL sequence, \( L(G) \) as a DOL language, \( LS(G) \) as a DOL length sequence and \( f_G \) as a DOL growth function.

We shall now briefly review several research areas concerning DOL systems.

I.1. DOL growth functions. Growth functions of DOL systems form a very natural object to investigate (also motivated by biological considerations which started the development of the \( L \) systems theory). The relationship between DOL length sequences and \( Z \)-rational sequences of numbers is by now quite well understood. Typical results here are:

**Theorem.** Assume that an \( N \)-rational sequence of numbers has a matrix representation \( u(n) = \pi M^n \eta \), \( n = 0, 1, 2, \ldots \), with either only positive entries in \( \pi \) or only positive entries in \( \eta \). Then \( u(n) \) is a DOL length sequence. \( \square \)

**Theorem.** Every \( Z \)-rational sequence can be expressed as the difference of two DOL length sequences. \( \square \)

Generating functions form a very useful tool in investigating DOL growth functions. The following result is typical in characterizing generating functions of DOL growth functions.

**Theorem.** A rational function \( F(x) \) with integral coefficients and written in lowest terms is the generating function of a DOL growth function not identical to the zero function if and only if either \( F(x) = a_0 + a_1 x + \ldots + a_n x^n \) where \( a_0, a_1, \ldots, a_n \) are positive integers, or else \( F(x) \) satisfies each of the following conditions:

(i) The constant term of its denominator equals 1.

(ii) The coefficients of the Taylor expansion \( F(x) = \sum_{n=0}^{\infty} a_n x^n \) are positive integers and, moreover, the ratio \( a_{n+1}/a_n \) is bounded by a constant.

(iii) Every pole \( x_0 \) of \( F(x) \) of the minimal absolute value is of the form \( x_0 = re^{\epsilon} \) where \( r = |x_0| \) and \( \epsilon \) is a root of unity. \( \square \)
I.2. Locally catenative DOL systems. A very natural way to generalize linear homogeneous recurrence relations to words is as follows.

A locally catenative formula (LCF for short) is an ordered $k$-tuple $v = (i_1, \ldots, i_k)$ of positive integers where $k > 2$ (we refer to $k$ as the width of $v$ and to $\max \{i_1, \ldots, i_k\}$ as the depth of $v$). An infinite sequence of words $\omega_0, \omega_1, \omega_2, \ldots$ satisfies $v$ with a cut $p \geq \max \{i_1, \ldots, i_k\}$ if, for all $n > p$, $\omega_{n} = \omega_{n-i_1} \cdots \omega_{n-i_k}$. A sequence of words satisfying some LCF $v$ with some cut is called $(v)$ locally catenative. A DOL system $G$ is called $(v)$-locally catenative if $E(G)$ is $(v)$-locally catenative. We say that $G$ is locally catenative of depth $d$ if $G$ is $v$-locally catenative for some LCF $v$ with depth of $v$ equal to $d$.

First of all we get the following correspondence between locally catenative DOL sequences and languages.

**Theorem.** A DOL system $G$ is locally catenative if and only if $L(G)^*$ is a finitely generated monoid.

The following result illustrates the relationship between a global property of a DOL sequence (namely its locally catenative property) and a local property of the underlying DOL system (namely the way its homomorphism is defined). Let $G = (\Sigma, h, \omega)$ be a DOL system with $\omega \in \Sigma$ where for no $a$ in $\Sigma$, $h(a) = \Lambda$. The graph of $G$, denoted $\mathcal{G}(G)$, is an ordered graph the nodes of which are elements of $\Sigma$ and, for $a, b \in \Sigma$, $(a, b)$ is an edge in $\mathcal{G}(G)$ if and only if $h(a) = ab\beta$ for some $\alpha, \beta \in \Sigma^*$.

**Theorem.** If there exists $a \in \Sigma$ such that $h^n(\omega) = a$ for some $n > 0$ and every cycle in $\mathcal{G}(G)$ goes through $a$ then $G$ is locally catenative.

The most important open problem concerning locally catenative DOL systems is the decidability status of the question: “Is an arbitrary DOL system locally catenative?” The best known result in this direction is:

**Theorem.** (1) It is decidable whether or not an arbitrary DOL system is locally catenative of depth $d$, where $d$ is an arbitrary positive integer. (2) It is decidable whether or not an arbitrary DOL system is locally catenative of width $d$, where $d$ is an arbitrary positive integer larger than 1.

I.3. DOL equivalence problem. One of the more challenging problems in the theory of DOL systems is the DOL sequence (respectively language) equivalence problem: “Given two arbitrary DOL systems $G_1, G_2$ is it decidable whether or not $E(G_1) = E(G_2)$ (respectively $L(G_1) = L(G_2)$)?”

The problem was solved quite recently (by K. Culik and I. Fris).

**Theorem.** The DOL sequence and language equivalence problems are decidable.

Various efforts to solve the above mentioned problems created quite a number of notions and results which are of interest also in traditional formal language theory.
For example one gets the following representation theorem for recursively enumerable languages.

**Theorem.** Let $K$ be a recursively enumerable language. There exist an alphabet $\Sigma$, homomorphisms $g, h_1, h_2$ and a regular language $M$ such that $K = g(\operatorname{Eq}(h_1, h_2) \cap M)$, where $\operatorname{Eq}(h_1, h_2) = \{x \in \Sigma^+: h_1(x) = h_2(x)\}$. □

**II. Single finite substitutions iterated.** A natural way to generalize DOL systems is to consider the iteration of a finite substitution rather than a homomorphism. (The difference between a finite substitution and a homomorphism is that the former maps each letter of the alphabet into a finite set of words whereas the latter maps each letter into a single word.) Since in such a case the generated language (rather than the sequence) becomes the primary concept, one therefore considers (as usual in formal language theory) an additional (terminal) alphabet.

An EOL system is a construct $G = (\Sigma, h, \omega, A)$ where $\Sigma, A$ are finite alphabets, $\Delta \subseteq \Sigma$, $\omega \in \Sigma^+$ and $h$ is a finite substitution from $\Sigma^*$ into $2^{\omega \star}$. The language of $G$ is defined by $L(G) = \{x \in A^*: x \in h \circ \omega \}$ for some $n > 0$. $L(G)$ is referred to as an EOL language.

The following results illustrate the combinatorial structure of EOL languages. They are especially useful for proving in general that various “concrete” languages are not EOL languages (which is often a difficult task). Let $K$ be a language over $\Sigma$ and let $B$ be a nonempty subset of $\Sigma$. Let $\mathcal{N}(K, B) = \{n: (\exists x)_{K}(\#_B x = n)\}$. We say that $B$ is numerically dispersed in $K$ if $\mathcal{N}(K, B)$ is infinite and, for every natural number $k$, there exists a natural number $n_k$ such that whenever $u_1$ and $u_2$ are in $\mathcal{N}(K, B)$ and $u_1 > u_2 > n_k$ then $u_1 - u_2 > k$. $B$ is clustered in $K$ if $\mathcal{N}(K, B)$ is infinite and there exist natural numbers $k_1, k_2$ both larger than 1 such that whenever a word $x$ in $K$ satisfies $\#_B x > k_1$ then $x$ contains at least two occurrences of letters from $B$, which lie at a distance smaller than $k_2$ from each other.

**Theorem.** Let $K$ be an EOL language over $\Sigma$ and $B$ a nonempty subset of $\Sigma$. If $B$ is numerically dispersed in $K$ then $B$ is clustered in $K$. □

Let $K$ be a language over an alphabet $\Sigma$ and let $B$ be a nonempty subset of $\Sigma$. We say that $K$ is $B$-determined if for every positive integer $k$ there exists a positive integer $n_k$ such that for every $x, y$ in $K$ if $|x|, |y| > n_k$, $x = x_1 u x_2$, $y = y_1 v x_2$ and $|u|, |v| < k$ then $\operatorname{Pres}_B(u) = \operatorname{Pres}_B(v)$.

**Theorem.** Let $K$ be a $B$-determined EOL language. There exist positive integer constants $c$ and $d$ such that, for every $x \in K$, if $\#_B x > c$ then $|x| < d^k n^x$. □

**Theorem.** Let $K$ be an EOL language over an alphabet $\Sigma$. If $K$ is $\Sigma$-determined then there exists a constant $c$ such that, for every nonnegative integer $n$, $\pi_n(K) < c n^3$. □

**III. Several homomorphisms iterated.** The language of a DOL system is obtained by applying to a fixed word an arbitrary homomorphism from the semigroup gene-
rated by a single homomorphism. Semigroups generated by a finite number of homomorphisms form a natural next step.

A DTOL system is a construct $G = (\Sigma, H, \omega)$ where $\Sigma$ is a finite alphabet, $\omega \in \Sigma^+$ and $H$ is a finite set of homomorphisms from $\Sigma^*$ into $\Sigma^*$. The language of $G$ is defined by $L(G) = \{ x \in \Sigma^* : x = h_n \ldots h_1(\omega) \text{ for } n \geq 0, h_i \in H \}$.

The following result describes a rather basic property of the set of all subwords of a DTOL language.

**Theorem.** Let $\Sigma$ be a finite alphabet such that $\# \Sigma = n \geq 2$. If $K$ is a DTOL language over $\Sigma$ then $\lim_{t \to \infty} (n^t(K)/n^t) = 0$. □

Adding an extra (terminal) alphabet one can define a richer class of languages. An EDTOL system is a construct $G = (\Sigma, H, \omega, \Lambda)$ where $(\Sigma, H, \omega)$ is a DTOL system and $\Lambda \subseteq \Sigma$. The language of $G$ is defined by $L(G) = \{ x \in \Lambda^*: x = h_n \ldots h_1(\omega) \text{ for } n \geq 0, h_i \in H \}$; it is referred to as an EDTOL language.

The following two results are very useful results on the combinatorial structure of EDTOL languages.

A function $f$ from $\mathbb{R}^+ \rightarrow \mathbb{R}^+$ is called slow if for every $a \in \mathbb{R}^+$ there exists $n_a \in \mathbb{R}^+$ such that for every $x \in \mathbb{R}^+$ if $x > n_a$ then $f(x) < x^a$.

Let $\Sigma$ be a finite alphabet and let $f : \mathbb{R}^+ \rightarrow \mathbb{R}^+$. A word $w$ over $\Sigma$ is called $f$-random if every two disjoint subwords of $w$ which are longer than $f(|w|)$ are different.

**Theorem.** For every EDTOL language $K$ and for every slow function $f$ there exists a constant $s$ such that for every $f$-random word $x$ in $K$ longer than $s$ there exist a positive integer $t$ and words $w_1, \ldots, w_t \in \Sigma^*$ such that $x = x_1 \omega w_1 x_2 \omega w_2 \ldots x_t \omega w_t$ for some $x_1, x_t \in \Sigma^*$ and, for every nonnegative integer $n$, $x_1 x_2 \omega \ldots x_t x_{t+1} \omega \ldots x_1$ is in $L$. □

**Theorem.** Let $K$ be an EDTOL language over an alphabet $\Sigma$, where $\# \Sigma = n \geq 2$. If Length$(K)$ does not contain an infinite arithmetic progression then

$$\lim_{l \to \infty} \frac{\# \{ w \in K : |w| = l \}}{n^l} = 0.$$ □

**IV. Several finite substitutions iterated.** In the same way as one generalized DOL systems to EOL systems one extends DTOL systems to obtain ETOL systems.

An ETOL system is a construct $G = (\Sigma, H, \omega, \Lambda)$ where $\Sigma$ is a finite alphabet, $\omega \in \Sigma^+$, $\Lambda \subseteq \Sigma$ and $H$ is a finite set of finite substitutions from $\Sigma^*$ into $2^{\Sigma^*}$. The language of $G$ is defined by $L(G) = \{ x \in \Lambda^* : x = h_n \ldots h_1(\omega) \text{ for } n \geq 0, h_i \in H \}$.

Here is a typical result concerning combinatorial structure of ETOL languages.

**Theorem.** Let $K$ be an ETOL language over an alphabet $\Sigma$. Then for every nonempty subset $\Delta$ of $\Sigma$ there exists a positive integer $k$ such that for every $x$ in $K$ either (i) $|\text{Pres}_\Delta x| < 1$, or (ii) there exists $a, b \in \Delta$ and $w \in \Sigma^*$ such that $x = x_1 awbx_2$ for some $x_1, x_2$ in $\Sigma^*$ with $|awb| < k$, or (iii) there exists an infinite subset $M$ of $K$ such that, for every $y$ in $M$, $|\text{Pres}_\Delta y| = |\text{Pres}_\Delta x|$. □
The following result is a typical "bridging" result. It allows one to construct examples of non ETOL languages providing that one has examples of languages that are not EDTOL.

**Theorem.** Let $\Sigma_1, \Sigma_2$ be two disjoint alphabets and let $K_1 \subseteq \Sigma^*_1, K_2 \subseteq \Sigma^*_2$. Let $f$ be a surjective function from $K_1$ into $K_2$ and let $K = \{wf(w) : w \in K_1\}$. Then

1. If $K$ is an ETOL language then $K_2$ is an EDTOL language.
2. If $f$ is a bijection, then also $K_1$ is an EDTOL language (if $K$ is an ETOL language). □

We hope that through this short note the reader acquires a taste of the theory of $L$ systems—a new and rapidly developing area of discrete mathematics with interesting connections to computer science and biology.

**References**


**Current address:** University of Leiden
2300 RA Leiden, The Netherlands