CODINGS OF OL LANGUAGES +

A. Ehrenfeucht
Department of Computer Science
University of Colorado, Boulder, Colorado

G. Rozenberg
Department of Mathematics
Utrecht University
Utrecht, The Netherlands

Report #CU-CS-025-73 August, 1973

All correspondence to:

G. Rozenberg
Institute of Mathematics
Utrecht University
Utrecht - Uithof
The Netherlands

+ This work supported by National Science Foundation Grant # GJ-660

ABSTRACT.

It is proved that a language is a coding (a letter-to-letter homomorphism) of a 0L language, if, and only if, it is an E0L language.

0. INTRODUCTION.

Developmental systems and languages are under active investigation and they are the subject of quite a large number of papers in both formal language theory and theoretical biology (see, e.g., [Herman and Rozenberg], [Lindenmayer], [Lindenmayer and Rozenberg], and their references).

EOL systems and languages form an interesting and intensively studied class within this theory (see, e.g., [Herman], [Herman and Rozenberg], [Rozenberg]).

An EOL system consists of:

(i) the finite alphabet Σ divided into the terminal (Σ_T) and nonterminal (Σ_N) alphabets,

(ii) the distinguished word, ω, over Σ,

(iii) the finite set of productions P,

which tell us by what string over Σ an occurrence of a symbol in a word may be replaced to get the "next word".

The language of an EOL system consists of all these strings over Σ_T which can be "derived" in a finite number of steps from ω in such a way that to get the next string (y) from the given one (x) one replaced each occurrence of each symbol in x according to a production from P.

A OL system (see, e.g.,[Rozenberg and Doucet]) is such an EOL system in which the alphabet of nonterminals is the empty set ($\Sigma_N = \emptyset$). Thus the only difference between the "generation of languages" by EOL and OL systems is the use of nonterminal symbols.

The OL systems are very natural models of biological development of some classes of organisms (for a discussion see [Lindenmayer]). The main reason for which EOL systems are interesting, as far as their biological motivation is concerned, is the fact that the class of languages generated by EOL systems contain the homomorphic images of the class of languages generated by OL systems (for a detailed discussion see, e.g., [Herman], [Lindenmayer and Rozenberg]).From the formal language theory point of view the question whether defining languages by EOL systems (thus using nonterminals) is equivalent to defining languages by EOL systems without nonterminals but, instead, taking homomorphic images of resulting languages, is very interesting on its own. An answer to this question would contribute to our understanding of the role different components of "grammar-like systems" play in defining languages.

For these reasons the problem of the relationship between the class of languages generated by EOL systems and the class of languages which are homomorphic images of languages generated by OL systems has been under active investigation in the theory for quite a long time.

In this paper we prove that the class of languages which are codings (letter-to-letter honomorphisms) of languages generated by OL systems is identical to the class of languages generated by EOL systems.

In other words, rather than to use nonterminal symbols (which, e.g., rarely have any biological interpretation) it is sufficient to generate a language by a OL system and then "translate" each word in this language using (only once) a coding table which is simply a letter-to-letter correspondence.

A number of applications of this result are also given.

1. PRELIMINARIES.

We assume the reader to be familiar with the basics of formal language theory, e.g., in the scope of [Ginsburg], whose notation and terminology we shall mostly follow. In addition to this we shall use the following notation:

(i) N denotes the set of non-negative integers and $N^+ = N - \{0\}$.

(ii) If x is a word, then $|x|$ denotes its length and Min{x} denotes the set of letters which occur in x.

(iii) If A is a set then 2^A denotes the set of subsets of A and, in the case when A is finite, #A denotes its cardinality.

(iv) If A is an ultimately periodic set of non-negative integers then thres (A) denotes the smallest integer j for which there exists a positive integer q such that, for every $i \geq j$, if i is in A then $(i + q)$ is in A. The smallest positive integer p such that, for every $i \geq$ thres (A), whenever i is in A then also $i + p$ is in A, is denoted by per (A).

(v) $\emptyset$ denotes the empty set and Λ denotes the empty word.

(vi) If $\mathcal{A}$ is a finite automaton, then $L(\mathcal{A})$ denotes its language.

(vii) A coding is a letter-to-letter homomorphism.

2. DEFINITIONS.

In this section we give basic definitions concerning developmental languages, which are relevant for this paper.

Definition 1. An ETOL system is a construct $G = \langle V_N, V_T, \mathcal{P}, \omega \rangle$ such that

V_N is a finite alphabet (of nonterminal letters or symbols),

V_T is a finite nonempty alphabet (of terminal letters or symbols), such that $V_N \cap V_T = \emptyset$,

ω is an element of $(V_N \cup V_T)^+$ (called the axiom of G),

$\mathcal{P}$ is a finite nonempty family, each element of which a finite nonempty set of the form $\{a \to \alpha : a$ is in $V_N \cup V_T$ and α is in $(V_N \cup V_T)^*\}$ (where we assume that the symbol $\to$ is not in $V_N \cup V_T$).

Each element P of $\mathcal{P}$ (called a table of G) satisfies the condition:

for every a in $V_N \cup V_T$ there exists at least one α in $(V_N \cup V_T)^*$ such that $a \to \alpha$ is in P.

(If P is in $\mathcal{P}$, and $a \to \alpha$ is in P then $a \to \alpha$ is called a production in P, or a production of G. We often write $a \underset{P}{\Rightarrow} \alpha$ for "$a \to \alpha$ is in P).

Definition 2. Let $G = \langle V_N, V_T, \mathcal{P}, \omega \rangle$ be an ETOL system.

(i) A production $a \to \alpha$ of G such that $\alpha = \Lambda$ is called an erasing production. G is called propagating if no production of G is an erasing production.

(ii) G is called deterministic if, for every P in $\mathcal{P}$ and every a in $V_N \cup V_T$, there exists exactly one α in $(V_N \cup V_T)^*$ such that $a \underset{P}{\to} \alpha$.

(We shall use letters P and D to denote the propagating and deterministic restrictions, respectively. For example, "an EDTOL system" means a deterministic ETOL system, and "an EDPTOL system" means a deterministic, propagating ETOL system.)

(iii) G is called an <u>EOL system</u>, if $\#\mathcal{P} = 1$. (In this case, if $\mathcal{P} = \{P\}$, then we often write G as $\langle V_N, V_T, P, \omega\rangle$.)

(iv) G is called a <u>OL system</u>, if $\#\mathcal{P} = 1$ and $V_N = \emptyset$. (In this case, if $\mathcal{P} = \{P\}$, then we often write G as $\langle V_T, P, \omega\rangle$.)

<u>Definition 3</u>. Let $G = \langle V_N, V_T, \mathcal{P}, \omega\rangle$ be an ETOL system.

(i) Let $x \in (V_N \cup V_T)^+$, say $x = b_1 b_2 \ldots b_t$ for some $b_2, b_2, \ldots, b_t$ in $V_N \cup V_T$, and let $y \in (V_N \cup V_T)^*$. We say that x <u>directly derives</u> y (<u>in</u> G), denoted as $x \underset{G}{\Rightarrow} y$, if there exist P in $\mathcal{P}$ and a sequence $\pi_1, \ldots, \pi_t$ of productions from P such that, for every i in $\{1, \ldots, t\}$, $\pi_i = b_i \to \alpha_i$ and $y = \alpha_1 \ldots \alpha_t$. (In this case we also write $x \underset{P}{\Rightarrow} y$, and we say that x <u>directly derives</u> y (<u>in</u> G) <u>using</u> P.)

(ii) Let $x \in (V_N \cup V_T)^+$ and $y \in (V_N \cup V_T)^*$. We say that x <u>derives</u> y <u>in</u> G, denoted as $x \overset{*}{\underset{G}{\Rightarrow}} y$, if,

either $x = y$,

or, for some $n > 0$, there exists a sequence $x_0, x_1, \ldots, x_n$ of words in $(V_N \cup V_T)^*$, such that $x_0 = x$, $x_n = y$ and $x_{i-1} \underset{G}{\Rightarrow} x_i$ for $1 \le i \le n$. (If the latter holds then we also write $x \overset{+}{\underset{G}{\Rightarrow}} y$. Also, if $\tau = T_1 \ldots T_n$ is a sequence of tables from $\mathcal{P}$ and $x_0 \underset{T_1}{\Rightarrow} x_1 \underset{T_2}{\Rightarrow} x_2 \ldots \underset{T_n}{\Rightarrow} x_n$, then we write $x \overset{\tau}{\underset{G}{\Rightarrow}} y$.)

(iii) For x in $(V_N \cup V_T)^+$ and y in $(V_N \cup V_T)^*$, a <u>derivation of</u> y <u>from</u> x (<u>in</u> G) is a sequence $D = (x_0, x_1, \ldots, x_n)$ of words in $(V_N \cup V_T)^*$ such that, for $1 \le i \le n$, $x_{i-1} \underset{G}{\Rightarrow} x_i$. If $x = \omega$, then D is called a <u>derivation of</u> y (<u>in</u> G), and if $x = \omega$ and $y \in V_T^*$ then D is called a <u>terminal derivation</u> (<u>in</u> G). If D is terminal, then we say that x <u>is derived in</u> G <u>in</u> n <u>steps</u>, and we write $x_0 \overset{n}{\underset{G}{\Rightarrow}} x_n$. (By definition $x \overset{0}{\underset{G}{\Rightarrow}} x$, for every x in $(V_N \cup V_T)^*$).

(iv) If G is an EOL system, say $\mathcal{P} = \{P\}$, and $D = (x_0, \ldots, x_n)$ is a derivation of x_n from x_0 in G, then the sequence $P_1, \ldots, P_n$ of subsets of P such that, for $1 \leq i \leq n$, $x_{i-1} \underset{G}{\Longrightarrow} x_i$ "using" all and only productions from P_i is called a control sequence of D.

ETOL systems are used to define languages as follows.

Definition 4. Let $G = \langle V_N, V_T, \mathcal{P}, \omega \rangle$ be an ETOL system. The language of G, denoted as L(G), is defined by $L(G) = \{x \in V_T^* : \omega \overset{*}{\underset{G}{\Longrightarrow}} x\}$.

Definition 5. A nonempty language K different from $\{\Lambda\}$ is called an ETOL (EOL, OL, EDTOL, etc.) language if, and only if, there exists an ETOL (EOL, OL, EDTOL, etc.) system G such that L(G) = K. A language K is called a COL language (a HOL language) if, and only if, there exists a OL language $\overline{K}$ and a coding h (a homomorphism h) such that $h(\overline{K}) = K$.

In the sequel $\mathcal{F}_{EOL}$, $\mathcal{F}_{COL}$ and $\mathcal{F}_{HOL}$ will denote the classes of EOL, COL and HOL languages, respectively.

Remark 1. In the sequel, given an ETOL system $G = \langle V_N, V_T, \mathcal{P}, \omega \rangle$, we shall sometimes consider $\mathcal{P}$ as a set of tables as defined in Definition 1, and sometimes we shall consider $\mathcal{P}$ to be the set of symbols ("names" of tables), but this should not lead to a confusion. (For example, we may talk about an alphabet $\mathcal{P}$, words over $\mathcal{P}$, etc.).

Remark 2. It is well-known (see, e.g., Rozenberg) that for every ETOL system G there exists an ETOL system $H = \langle V_N, V_T, \mathcal{P}, \omega \rangle$ such that ω is in V_N. Thus in the sequel we shall often assume that an ETOL system we deal with is such that its axiom is a nonterminal symbol, and in this case the axiom shall be denoted by the symbol S.

3. SPECTRA OF SYMBOLS IN EOL SYSTEMS

In this section we introduce the basic notion of the so-called spectrum of a symbol in an EOL system and prove its basic property. (In this and the next section we shall deal only with EPOL and EDPTOL systems, and although most of the definitions would carry over to the more general classes of EOL and EDTOL systems, for the sake of clarity, we state them in restricted versions only.)

We start with classifying symbols in ETOL systems.

Definition 6. Let $G = \langle V_N, V_T, \mathcal{P}, S\rangle$ be an EPTOL system.

(i) A letter a in $V_N \cup V_T$ is called alive if there exists a positive integer k and a word w in V_T^+ such that $a \overset{k}{\underset{G}{\Longrightarrow}} w$.

(ii) An alive letter a is called vital if for every k in N^+ there exists an ℓ in N^+ such that $\ell > k$ and $a \overset{\ell}{\underset{G}{\Longrightarrow}} w$, for some w in V_T^+. (A_G will denote the set of all alive symbols from $V_N \cup V_T$).

The next notion is the central one for this paper.

Definition 7. Let $G = \langle V_N, V_T, \mathcal{P}, S\rangle$ be an EPTOL system and let a be in $V_N \cup V_T$. The spectrum of a in G, denoted as Spec (G, a), is defined by

$$\text{Spec}\,(G, a) = \{n \in N^+ : a \overset{n}{\underset{G}{\Longrightarrow}} w \text{ for some } w \text{ in } V_T^+\}.$$

Thus a positive integer n is in Spec (G, a) if, and only if, a can derive (in G) in n steps a word consisting of terminal symbols only.

We shall prove now that spectra of letters in EOL systems are ultimately periodic sets.

Lemma 1. If $G = \langle V_N, V_T, P, S\rangle$ is an EDPTOL system, then, for every a in $V_N \cup V_T$, Spec (G, a) is an ultimately periodic set.

Proof.

Let $G = \langle V_N, V_T, P, S\rangle$ be an EDPTOL system and let a be in $V_N \cup V_T$.

Let $A(G, a) = \langle Q, \Sigma, \delta, q_0, F\rangle$ be a finite automaton such that

$Q = 2^{V_N \cup V_T} - \emptyset$,

Σ is a one-letter alphabet, say $\Sigma = \{\sigma\}$,

$q_0 = \{a\}$,

$F = \{q \in Q : q \subseteq V_T\}$, and

for every $q, \bar{q}$ in Q, $\delta(q, \sigma) = \bar{q}$ if, and only if, there exist P in P and x, y in $(V_N \cup V_T)^+$, such that Min(x) = q, Min(y) = $\bar{q}$ and $x \underset{P}{\Longrightarrow} y$.

From the construction of $A(G, a)$ it immediately follows that $a \overset{n}{\underset{G}{\Longrightarrow}} w$, for some n in N^+ and w in V_T^*, if, and only if, σ^n is in $L(A(G, a))$. Thus $\mathrm{Spec}(G, a) = \{n : \sigma^n \text{ is in } L(A(G, a))\}$.

But it is well-known (see, e.g., [Ginsburg], Theorem 2.1.2) that if T is a finite automaton over a one-symbol alphabet, say $\{\sigma\}$, then $\{n : \sigma^n \text{ is in } L(A)\}$ is an ultimately periodic set.

Consequently, Spec(G, a) is an ultimately periodic set, and Lemma 1 holds.

Lemma 2. For every EPOL system $G = \langle V_N, V_T, P, S\rangle$ there exists an EDPTOL system $H = \langle V_N, V_T, P, S\rangle$ such that, for every a in $V_N \cup V_T$, Spec(G, a) = Spec(H, a).

Proof.

Let $G = \langle V_N, V_T, P, S\rangle$ be an EPTOL system, where $V_N \cup V_T = \{\sigma_1, \ldots, \sigma_n\}$.

Let $H = \langle V_N, V_T, \mathcal{P}, S\rangle$ be an EDPTOL system, where a table $\{\sigma_1 \to \alpha_1,\dots,\sigma_n \to \alpha_n\}$ is in $\mathcal{P}$ if, and only if, it is a subset of P.

Let a be in $V_N \cup V_T$.

It is clear that every derivation in H is also a derivation in G, hence $\mathrm{Spec}(H, a) \subseteq \mathrm{Spec}(G, a)$.

Let n be in Spec(G, a), $D = (a,x_1,\dots,x_n)$ be a terminal derivation from a in G and $(Z_1,\dots,Z_n)$ be a control sequence of D. Let $P_1,\dots,P_n$ be a sequence of tables from $\mathcal{P}$ such that, for every i in $\{1,\dots,n\}$, $P_i \subseteq Z_i$. If $D' = (a, y_1,\dots,y_n)$ is a derivation in H such that $a \underset{P_1}{\Longrightarrow} y_1$ and $y_j \underset{P_j}{\Longrightarrow} y_{j+1}$ for every j in $\{1,\dots,n-1\}$, then obviously $\mathrm{Min}(y_i) \subseteq \mathrm{Min}(x_i)$ for every i in $\{1,\dots,n\}$. Consequently, y_n is in V_T^* and n is in Spec(H, a). Thus $\mathrm{Spec}(G, a) \subseteq \mathrm{Spec}(H, a)$.

Thus, for every a in $V_N \cup V_T$, Spec(G, a) = Spec(H, a). As it is also clear that H is propagating, Lemma 2 holds.

<u>Lemma 3</u>. If $G = \langle V_N, V_T, P, S\rangle$ is an EOL system, then, for every a in $V_N \cup V_T$, Spec(G, a) is an ultimately periodic set.

<u>Proof</u>.

This result follows immediately from Lemma 1 and Lemma 2.

4. THE CASE OF EP0L LANGUAGES.

In this section we show that every EP0L language is a coding of a 0L language.

First we need the following notion and a construct.

Definition 8. Let $G = \langle V_N, V_T, P, S\rangle$ be an EP0L system. A uniform period of G, denoted as m_G, is the smallest positive integer such that

(i) for every a in $(V_N \cup V_T) - A_G$, if $a \overset{k}{\underset{G}{\Longrightarrow}} w$ for some $k \geq m_G$ and w in $(V_N \cup V_T)^*$, then w is not in $(V_N \cup V_T)^*$,

(ii) for every a in A_G, $m_G >$ thres(Spec(G, a)) and per(Spec(G, a)) divides m_G.

Construction. Let $G = \langle V_N, V_T, P, S\rangle$ be an EP0L system and let k be a non-negative integer smaller than m_G. Let

$Ax(G, k) = \{w \in A_G^+ : S \overset{m_G}{\underset{G}{\Longrightarrow}} w$ and, for every a in Min(w), $m_G + k$ is in Spec(G, a)$\}$.

If Ax(G, k) is not empty, then, for every w in Ax(G, k), we define a 0L system $G(k, w) = \langle \Sigma_{k,w}, R_{k,w}, w\rangle$ as follows:

(i) $\Sigma_{k,w} = \{a \in A_G : m_G + k$ is in Spec(G, a) and, for some $\ell \geq 0$, a is in Min(y) for some y such that $w \overset{\ell \cdot m_G}{\underset{G}{\Longrightarrow}} y\}$,

(ii) for every a in $\Sigma_{k,w}$ and α in $\Sigma_{k,w}^+$, $a \to \alpha$ is in $R_{k,w}$, if and only if, $a \overset{m_G}{\underset{G}{\Longrightarrow}} \alpha$.

We define M(G(k, w)) by

$M(G(k, w)) = \{x \in V_T^+ :$ there exists a word y in L(G(k, w)) such that $y \overset{m_G+k}{\underset{G}{\Longrightarrow}} x\}$.

We shall show now that, for the given EP0L system G, the union of M(G(k, w)) over all $k < m_G$ and w in Ax(G, k) is identical (modulo a finite set) to L(G).

<u>Lemma 4</u>. Let $G = \langle V_N, V_T, P, S \rangle$ be an EPOL system. Let $F_G = \{w \in V_T^+ : S \overset{\ell}{\underset{G}{\Longrightarrow}} w \text{ for some } \ell < 2m_G\}$. Then

$$F_G \cup \bigcup_{k < m_G} \bigcup_{w \in Ax(G, k)} M(G(k, w)) = L(G).$$

<u>Proof</u>.

Obviously, $F_G \cup \bigcup_{k < m_G} \bigcup_{w \in Ax(G, k)} M(G(k, w)) \subseteq L(G)$.

Now, let us assume that x is in L(G).

If x can be derived in G in less than $2m_G$ steps, then x is in F_G and consequently x is in $F_G \cup \bigcup_{k < m_G} \bigcup_{w \in Ax(G, k)} M(G(k, w))$.

Thus, let us assume that x is in L(G) and x is derived in G in at least $2m_G$ steps. Let $D = (S, x_1, \ldots, x_{m_G}, \ldots, x_p = x)$ be a derivation of x in G where $p \geq 2m_G$ and $p = \ell_p \cdot m_G + k_p$ for some ℓ_p in N^+ and k_p in $\{0, \ldots, m_G - 1\}$.

If ℓ is in $\{1, \ldots, \ell_p - 1\}$ and a is in $Min(x_{\ell \cdot m_G})$, then

(i) a is vital because $a \overset{t}{\underset{G}{\Longrightarrow}} x$ for some word x in V_T^+, where $t = (\ell_p \cdot m_G + k_p) - \ell \cdot m_G$ is not smaller than m_G.

(ii) $(\ell_p - \ell)m_G + k_p$ is in Spec(G, a) and so (because Spec(G, a) is an ultimately periodic set with period m_G and threshold smaller than m_G) $m_G + k_p$ is also in Spec(G, a).

Consequently, x is in $M(G(k_p, x_{m_G}))$ and so it is in $F_G \cup \bigcup_{k < m_G} \bigcup_{w \in Ax(G, k)} M(G(k_p, w))$.

Thus $F_G \cup \bigcup_{k < m_G} \bigcup_{w \in Ax(G, k)} M(G(k, w)) = L(G)$, and Lemma 4 holds.

Now we shall prove that each component language in the "union formula for L(G)" as given in the statement of Lemma 4 is a finite union of codings of OL languages.

Lemma 5. Let G be an EP0L system $k < m_G$, $Ax(G, k) \neq \emptyset$ and let w be in Ax(G, k). Then there exist a finite set $\{H_1,\ldots,H_f\}$ of 0L systems and a coding h such that

$$M(G(k, w)) = \bigcup_{i=1}^{f} h(H_i).$$

Proof.

Let $G = \langle V_N, V_T, P, S\rangle$ be an EP0L system, $k < m_G$ and $Ax(G, k) \neq \emptyset$. Let w be in Ax(G, k), $w = b_1,\ldots,b_t$ where b_i is in A_G for $1 \leq i \leq t$ and let $G(k, w) = \langle \Sigma_{k,w}, R_{k,w}, w\rangle$.

For every a in $\Sigma_{k,w}$ let $U(a, k) = \{w \in V_T^+ : a \overset{m_G+k}{\underset{G}{\Longrightarrow}} w\}$, say $U(a, k) = \{\alpha_{a,k,1}, \alpha_{a,k,2},\ldots, \alpha_{a,k,u(a,k)}\}$.

Let $\overline{\Sigma}_{k,w} = \{[a, i, b] : a \in \Sigma_{k,w}, i \in \{1,\ldots,u(a, k)\} \text{ and } b \in V_T\} \cup \{\overline{[a, i, b]} : a \in \Sigma_{k,w}, i \in \{1,\ldots,u(a, k)\} \text{ and } b \in V_T\}$.

Let $W(w) = \{[b_1, i_1, c_{11}][b_1, i_1, c_{12}]\ldots\overline{[b_1, i_1, c_{1r_1}]}[b_2, i_2, c_{21}]\ldots\overline{[b_2, i_2, c_{2r_2}]}\ldots[b_t, i_t, c_{t1}]\ldots\overline{[b_t, i_t, c_{tr_t}]} : i_j \in \{1,\ldots,u(b_j, k)\}$ for $1 \leq j \leq t$, $c_{11} c_{12}\ldots c_{1r_1} = \alpha_{b_1, k, i_1}$, $c_{21} c_{22}\ldots c_{2r_2} = \alpha_{b_2, k, i_2},\ldots, c_{t1} c_{t2}\ldots c_{tr_t} = \alpha_{b_t, k, i_t}\}$

Let $\overline{R}_{k,w} = \{[a, i, b] \to \Lambda : a \in \Sigma_{k,w}, i \in \{1,\ldots,u(a, k)\} \text{ and } b \in V_T\} \cup \{\overline{[a, i, b]} \to [c_1, i_1, d_{11}][c_1, i_1, d_{12}]\ldots\overline{[c_1, i_1, d_{1v_1}]}[c_2, i_2, d_{21}]\ldots\overline{[c_2, i_2, d_{2v_2}]}\ldots[c_s, i_s, d_{s1}]\ldots\overline{[c_s, i_s, d_{sv_s}]} : a \to c_1\ldots c_s$ is in $R_{k,w}$, $i_j \in \{1,\ldots,u(c_j,k)\}$ for $1 \leq j \leq s$, $d_{11}\ldots d_{1v_1} = \alpha_{c_1, k, i_1}$, $d_{21}\ldots d_{2v_2} = \alpha_{c_2, k, i_2},\ldots,d_s\ldots d_{sv_s} = \alpha_{c_s, k, i_s}\}$.

Let, for each z in W(w), G(k, w, z) be a 0L system such that $G(k, w, z) = \langle \overline{\Sigma}_{k,w}, \overline{R}_{k,w}, z\rangle$.

Finally, let h be a coding from $\overline{\Sigma}_{k,w}$ into V_T such that $h([a, i, b]) = h(\overline{[a, i, b]}) = b$.

We leave to the reader the obvious, but tedious, proof of the fact that

$$M(G(k, w)) = \bigcup_{z \in W} h(L(G(k, w, z))).$$

Thus Lemma 5 holds.

The following two results are obvious, so we give them without proofs.

Lemma 6. If K is a finite language, then there exists a OL system G and a coding h such that $K = h(L(G))$.

Lemma 7. If K is a language such that there exists a finite set $\{H_1,\ldots,H_f\}$ of OL systems and a finite set of codings $\{h_1,\ldots,h_f\}$ such that $K = \bigcup_{i=1}^{f} h_i(L(H_i))$, then there exist a OL system G and a coding $\bar{h}$ such that $K = h(L(G))$.

Now we can show that each EPOL language is a coding of a OL language.

Lemma 8. For every EPOL language K there exist a OL language L and a coding h such that $K = h(L)$.

Proof.

This result follows directly from Lemmata 4 through 7.

5. THE MAIN RESULT AND ITS APPLICATIONS.

In this section we prove that the classes $\mathcal{F}_{EOL}$, $\mathcal{F}_{COL}$ and $\mathcal{F}_{HOL}$ are identical.

We start with the following result, the obvious proof of which is left to the reader.

Lemma 9. If a language L is a coding of a OL language, then also the language $L \cup \{\Lambda\}$ is a coding of a OL language.

Now we can prove the following result.

Theorem 1. For every EOL language K there exist a OL language L and a coding h such that $K = h(L)$.

Proof.

This theorem follows from Lemma 9 and from the known fact (see, e.g., [Herman]) that for each EOL system G there exists an EPOL system H such that $L(G) - \{\Lambda\} = L(H)$.

This result together with previously known results gives us the main theorem of this paper.

Theorem 2 (the main result). $\mathcal{F}_{EOL} = \mathcal{F}_{COL} = \mathcal{F}_{HOL}$.

Proof.

This follows from Theorem 1, from the known fact (see, e.g., [Herman]) that the class of EOL languages is closed with respect to homomorphic mappings and from the fact that each coding is a homomorphism.

It is interesting to observe that the above result is an "algorithmic" result in the following sense (the next result is a sample of the three possible results of the same character).

Proposition 1. The classes F_{EOL} and F_{COL} are effectively equal meaning that

(i) there exists an algorithm which, given an arbitrary EOL system G produces a OL system H and a coding h such that L(G) = h(L(H)), and

(ii) there exists an algorithm which, given an arbitrary OL system G and a coding h produces an EOL system H such that L(H) = h(L(G)).

Proof.

This follows from the fact (the straightforward but tedious proof of which we leave to the reader) that the proof of Theorem 2 is effective, meaning that all constructions involved may be effectively performed.

Theorem 2 has a number of interesting corrollaries and applications. We will now give three of them.

As the first one we discuss a result relating to the sets of lengths of words of EOL and OL languages. Investigations of lengths of words generated by different classes of developmental systems is a well-motivated and an active research area in the theory of developmental systems and languages (see, e.g., [Paz and Salomaa]). The following theorem is a new result in this area.

Definition 9. If G is an ETOL system then the length set of G, denoted as Lg(G), is defined by $Lg(G) = \{n \in N : |w| = n \text{ for some } w \text{ in } L(G)\}$.

Theorem 3. A subset U of nonnegative integers is the length set of an EOL system if, and only if, it is the length set of a OL system.

Proof.

This result follows from Theorem 1, from the fact that every OL language is, by definition, an EOL language and from the fact that coding is a "length preserving mapping" meaning that if x is a word and h is a coding such that h(x) is defined, then $|x| = |h(x)|$.

As the second application of the main result we get the following (rather "technical") theorem which is a solution to an open problem stated in [Herman, Lindenmayer and Rozenberg]. (The familiarity with this paper is assumed now.)

Theorem 4. For every recurrence system S there exists a recurrence system $\overline{S}$ such that $\overline{S}$ is in nearly standard form and $L(S) = L(\overline{S})$.

Proof.

It is proven in [Herman, Lindenmayer and Rozenberg], that a language is generated by a recurrence system in nearly standard form if, and only if, it is a homomorphic image of a OL language. Thus Theorem 4 follows now from Theorem 2.

Finally we have a very simple proof of the following result due to K. Culik II (personal communications).

Theorem 5. Every context-free language is a coding of a OL language.

Proof.

It is well-known (see, e.g., [Herman and Rozenberg], Theorem 4.6) that every context-free language is an EOL language, and so, by Theorem 1, every context-free language is a coding of a OL language.

REFERENCES .

Ginsburg, S., The mathematical theory of context-free languages, McGraw-Hill, 1966.

Herman, G., Closure properties of some families of languages associated with biological systems, Information and Control, to appear.

Herman, G., Lindemayer, A., and Rozenberg, G., Description of developmental languages using recurrence systems, submitted for publication.

Herman, G., and Rozenberg, G., Developmental systems and languages, North-Holland Publ. Company, to appear.

Lindenmayer, A., Mathematical models for cellular interactions in development, Parts I and II, Journal of Theoretical Biology, 1968, v. 18, 280-315.

Lindenmayer, A., and Rozenberg, G., Developmental systems and languages, Proc. of the Fourth ACM Symposium on the Theory of Computing, 1972.

Paz, A., and Salamaa, A., Integral sequential word function and growth equivalence of Lindemayer systems, Information and Control, to appear.

Rozenberg, G., Extension of tabled OL systems and languages, International Journal of Computer and Information Sciences, to appear.

Rozenberg, G., and Doucet, P., On OL languages, Information and Control, 1971, v. 19, 302-318.

GR:cah