On Structure of Derivations
in Deterministic ETOL Systems*

by

A. Ehrenfeucht**

and

G. Rozenberg***

Report #CU-CS-046-74 July 1974

* This work supported by NSF Grant #GJ-660

** Department of Computer Science
University of Colorado
Boulder, Colorado 80302 U.S.A.

*** Institute of Mathematics
Utrecht University
Utrecht-De Uithof HOLLAND
and
Department of Mathematics
University of Antwerp, U.I.A.
Wilrijk BELGIUM

All correspondence to: G. Rozenberg
Institute of Mathematics
Utrecht-De Uithof
HOLLAND

ABSTRACT

This paper investigates the structure of derivations in deterministic ETOL systems. The main theorem says that in a deterministic ETOL system each derivation of a long enough word of a special kind (the so-called f-random word) has a very strong combinatorial structure. In fact the main result of this paper is very essential for proving useful properties of deterministic ETOL languages, which is demonstrated already in a number of papers.

INTRODUCTION

The theory of L systems became recently one of the most vigorously investigated areas of formal language theory. (The reader is referred to Herman and Rozenberg [6] and to Rozenberg and Salomaa [9] which are the most extensive sources of readings on the theory of L systems as of today.)

One of the central families of L languages (languages generated by L systems) is the family of ETOL languages (see, e.g., Christensen [1], Downey [2], Rozenberg [8] and Salomaa [10]). In turn the deterministic subfamily of the family of ETOL languages appears to be a very important one (see, e.g., Ehrenfeucht and Rozenberg [4]).

This paper investigates deterministic ETOL systems from the point of view of the structure of derivations in these systems. The main idea behind our approach can be described as follows.

The absolutely parallel way of rewriting in L systems (all occurrences of all letters in a string are being rewritten in a single derivation step) causes essential difficulties in investigating the structure of L languages via the structure of derivations in L systems. Take for example the famous Bar Hillel pumping lemma for context free languages. It holds because if a derivation in a context free grammar is long enough then one can always find a self-embedding letter (one which derives itself and something else) and then iterate the piece of the derivation concerned with rewriting of this particular letter with the rest of the string remaining the same. This does not work in L systems because the whole string must be rewritten in a single derivation step, and thus while, e.g., rewriting a self-embedding letter the rest of the string will (in general) be also changed.

One way of overcoming this particular obstacle is to consider derivations of the subset of the language generated by the system and then to introduce a much finer classification of symbols than self-embedding and non-self-embedding ones.

We have done this in this paper for the use of deterministic ETOL systems.

Thus this paper is intended as a contribution to the very important area of L systems theory: understanding the structure of a single L system and henceforth the structure of a single L language. Unless we understand the structure of a single L system (language) the theory of L systems will be missing a very important point.

In the second section we introduce basic definitions concerning ETOL systems and languages, and then in the third section we introduce various notions concerning derivations in deterministic ETOL systems. Section IV presents two technical results which are then used in Section V to prove the main result of this paper. Roughly speaking it says that each derivation of the so-called f-random word in an EDTOL system has a very definite "backbone structure" which goes through relatively many words of the derivation. This result is then discussed in Section VI.

II. BASIC DEFINITIONS

In this section we introduce the notions of an ETOL system and an ETOL language and illustrate them by examples.

Definition 1. An extended table L system without interactions, abbreviated as an ETOL system, is defined as a construct $G = \langle V, \mathcal{P}, \omega, \Sigma \rangle$ such that

1) V is a finite set (called the alphabet of G).

2) $\mathcal{P}$ in a finite set (called the set of tables of G), each element of which is a finite subset of $V \times V^*$. $\mathcal{P}$ satisfies the following (completeness) condition:

$$(\forall P)_{\mathcal{P}}(\forall a)_V(\exists \alpha)_{V^*}(\langle a, \alpha\rangle \varepsilon P).$$

3) $\omega \varepsilon V^+$ (called the axiom of G).

4) $\Sigma \subseteq V$ (called the target alphabet of G).

We assume that V, Σ and each P in $\mathcal{P}$ are nonempty sets.

Definition 2. An ETOL system $G = \langle V, \mathcal{P}, \omega, \Sigma \rangle$ is called

1) deterministic if for each P in $\mathcal{P}$ and each a in V there exists exactly one α in V^* such that $\langle a, \alpha \rangle$ is in P,

2) propagating if for each P in $\mathcal{P}$ we have $P \subseteq V \times V^+$.

We use letters D and P to denote the deterministic and the propagating restrictions respectively. Thus, for example, "an EDTOL system" means "a deterministic ETOL system" and "an EPDTOL system" means "a deterministic propagating ETOL system."

Definition 3. Let $G = \langle V, \mathcal{P}, \omega, \Sigma \rangle$ be an ETOL system. Let $x \varepsilon V^+$, $x = a_1 \cdots a_k$, where each a_j, $1 \leq j \leq k$, is an element of V and let $y \varepsilon V^*$. We say that x directly derives y in G (denoted as $x \underset{G}{\Rightarrow} y$) if and only if there exist P in $\mathcal{P}$ and $p_1, \ldots, p_k$ in P such that $p_1 = \langle a_1, \alpha_1 \rangle, \ldots, p_k = \langle a_k, \alpha_k \rangle$ and $y = \alpha_1 \cdots \alpha_k$. We say that x derives y in G (denoted as $x \overset{*}{\underset{G}{\Rightarrow}} y$)

if and only if either (i) there exists a sequence of words $x_0, x_1, \cdots, x_n$ in V* ($n > 1$) such that $x_0 = x_1$, $x_n = y$ and $x_0 \underset{G}{\Rightarrow} x_1 \underset{G}{\Rightarrow} x_2 \underset{G}{\Rightarrow} \cdots \underset{G}{\Rightarrow} x_n$ or (ii) $x = y$.

<u>Definition 4.</u> Let $G = \langle V, \mathcal{P}, \omega, \Sigma \rangle$ be an ETOL system. The language of G, denoted as L(G), is defined as $L(G) = \{x \in \Sigma^* : \omega \overset{*}{\underset{G}{\Rightarrow}} x\}$.

<u>Definition 5.</u> A language K is called an ETOL (EDTOL, EPDTOL) <u>language</u> if there exists an ETOL (EDTOL, EPDTOL) system G such that L(G) = K.

<u>Notation.</u> Let $G = \langle V, \mathcal{P}, \omega, \Sigma \rangle$ be an ETOL system. If $\langle a, \alpha \rangle$ is an element of some P in $\mathcal{P}$ then we call it a <u>production</u> and write $a \to \alpha$ is in P or $a \underset{P}{\to} \alpha$ or $P(a) = \alpha$.

We end this section with some examples of ETOL systems and languages.

<u>Example 1.</u> Let $G_1 = \langle V, \mathcal{P}, \omega, \Sigma \rangle$ where $V = \{A, B, a\}$, $\Sigma = \{a\}$, $\omega = AB$ and $\mathcal{P} = \{P_1, P_2\}$ where

$P_1 = \{A \to A^2, B \to B^3, a \to a\}$, $P_2 = \{A \to a, B \to a, a \to a\}$.

G_1 is an EPDTOL system where $L(G_1) = \{a^{2^n+3^n} : n > 0\}$.

<u>Example 2.</u> Let $G_2 = \langle \{a, b, A, B, C, D, F\}, \mathcal{P}, CD, \{a, b\} \rangle$, where $\mathcal{P} = \{P_1, P_2, P_3\}$ and

$P_1 = \{a \to F, b \to F, A \to A, B \to B, C \to ACB, D \to DA\}$,

$P_2 = \{a \to F, b \to F, A \to A, B \to B, C \to CB, D \to D\}$,

$P_3 = \{a \to F, b \to F, A \to a, B \to b, C \to \Lambda, D \to \Lambda\}$.

G_2 is an EDTOL system which is not propagating, and $L(G_2) = \{a^n b^m a^n : n \geq 0, m \geq n\}$.

III. DERIVATIONS AND SUBDERIVATIONS IN EDTOL SYSTEMS

To each word in the language of an ETOL system there corresponds a "derivation in G" which is a precise description how the word may be generated in G. In this paper we investigate the structure of derivations in EDTOL systems and this section introduces all necessary notions concerning this topic.

Definition 6. Let $G = \langle V, P, \omega, \Sigma \rangle$ be an EDTOL system. A derivation (of y from x) in G is a construct $D = ((x_0, \cdots, x_k), (T_0, \cdots, T_{k-1}), \mathcal{O})$ where $k \geqslant 2$ and

1) $x_0, \cdots, x_k$ are in V^*,
2) $T_0, \cdots, T_{k-1}$ are in P
3) $\mathcal{O}$ is an unambiguous description which tells us, for each j in $\{0, \cdots, k-1\}$, how each occurrence in x_j is rewritten using T_j to obtain x_{j+1},
4) $x_0 = x$ and $x_k = y$.

If $x = \omega$ then we simply say that D is a derivation (of y) in G.

Definition 7. Let $G = \langle V, P, \omega, \Sigma \rangle$ be an EDTOL system and let $D = ((x_0, \cdots x_k), (T_0, \cdots, T_{k-1}), \mathcal{O})$ be a derivation in G. For each occurrence a in x_j, $1 \leqslant j \leqslant k$, by a contribution of a in D, denoted as $\mathrm{Contr}_D(a)$, we mean the whole subword of x_k which is derived from a.

Definition 8. Let $G = \langle V, P, \omega, \Sigma \rangle$ be an EDTOL system and let $D = ((x_0, \cdots, x_k), (T_0, \cdots, T_{k-1}), \mathcal{O})$ be a derivation in G. A subderivation of D is a construct $\overline{D} = ((x_{i_0}, \cdots, x_{i_q}), (P_{i_0}, \cdots, P_{i_{q-1}}), \overline{\mathcal{O}})$ where

1) $0 \leqslant i_0 < i_1 < \cdots < i_q \leqslant k - 1$,
2) for each j in $\{0, \cdots, q-1\}$, $P_{i_j} = T_{i_j} T_{i_{j+1}} \cdots T_{i_{j+1}-1}$,
3) $\overline{\mathcal{O}}$ is an unambiguous description which tells us, for each j in $\{0, \cdots, q-1\}$, how each occurrence in x_{i_j} is rewritten by P_{i_j} to obtain $x_{i_{j+1}}$.

Remark

Although a subderivation of a derivation in G does not have to be a derivation in G we shall use for subderivations the same terminology as for derivations and this should not lead to confusion. (For example we talk about tables used in a subderivation). It is clear that to determine a subderivation $\bar{D}$ of a given derivation D it suffices to indicate which words of D form the sequence of words of $\bar{D}$. We will also talk about a subderivation $\bar{\bar{D}}$ of a subderivation $\bar{D}$ of D meaning a subderivation of D the words of which are chosen from the words of $\bar{D}$. (In this sense we have that a subderivation of a subderivation of a derivation D is a subderivation of the derivation D. Given a subderivation $\bar{D}$ of D and an occurrence a in a word of $\bar{D}$ we talk about $\mathrm{Contr}_{\bar{D}}(a)$ in an obvious sense.

In this paper we shall first investigate the structure of derivations in EPDTOL systems and then we shall discuss how the relaxing of the propagating restriction influences the structure of derivations.

Definition 9. Let $G = \langle V, P, \omega, \Sigma \rangle$ be an EPDTOL system and let f be a function from $\mathcal{R}_{pos}$ into $\mathcal{R}_{pos}$. Let D be a derivation in G and let $\bar{D} = ((x_0, \dots, x_k), (T_0, \dots, T_{k-1}), \mathcal{P})$ be a subderivation of $\bar{D}$. Let a be an occurrence (of A from V) in x_t for some t in $\{0, \dots, k\}$.

1) a is called (f,D)-big (in x_t), if $|\mathrm{Contr}_D(a)| > f(n)$,

2) a is called (f,D)-small (in x_t), if $|\mathrm{Contr}_D(a)| \leq f(n)$,

3) a is called unique (in x_t) if a is the only occurrence of A in x_t

4) a is called multiple (in x_t) if a is not unique (in x_t),

5) a is called $\bar{D}$-recursive (in x_t) if $T_t(a)$ contains an occurrence of A,

6) a is called $\bar{D}$-nonrecursive (in x_t) if a is not $\bar{D}$-recursive (in x_t).

Remark

1) Note that in an EDTOL system each occurrence of the same letter in a

word is rewritten in the same way during a derivation process. Hence we can talk about (f,D)-big (in x_t), (f,D)-small (in x_t), unique (in x_t), multiple (in x_t), $\bar{D}$-recursive (in x_t) and $\bar{D}$-nonrecursive (in x_t) letters.

2) Whenever f or D or $\bar{D}$ is fixed in considerations we will simplify the terminology in the obvious way (for example we can talk about big letters (in x_t) or about recursive letters (in x_t)).

Definition 10. Let $G = \langle V, P, \omega, \Sigma \rangle$ be an EPDTOL system and let f be a function from $\mathcal{R}_{pos}$ into $\mathcal{R}_{pos}$. Let D be a derivation in G and let $\bar{D} = ((x_0, \cdots, x_k), (T_0, \cdots, T_{k-1}), \bar{\mathcal{O}})$ be a subderivation of D. We say that $\bar{D}$ is neat (with respect to D and f) if the following holds:

1) $Min(x_0) = Min(x_1) = \cdots = Min(x_k)$.

2) If j is in $\{0, \cdots, k\}$ and A is a letter from $Min(x_j)$, then A is big (small, unique, multiple, recursive, nonrecursive) in x_j if and only if A is big (small, unique, multiple, recursive or nonrecursive respectively) in x_t for every t in $\{0, \cdots, k\}$.

3) For every j in $\{0, \cdots, k\}$, $Min(x_j)$ contains a big recursive letter.

4) For every j in $\{0, \cdots, k\}$ and every A in $Min(x_j)$, if A is big then A is unique.

5) For every j in $\{0, \cdots, k-1\}$

5.1) T_j contains a production of the form $A \to \alpha$ where A is a big letter and α contains small letters, and

5.2) If $A \to \alpha$ is in T_j, then

if A is small recursive, then $\alpha = A$, and

if A is nonrecursive then α consists of small recursive letters only.

6) For every i, j in $\{0, \cdots, k\}$ and every A in V, if a is a small occurrence of A in x_2 and b is a small occurrence of A in x_j then $|Contr_D(a)| = |Contr_D(b)|$.

7) For every big recursive letter A and for every i, j in 0, . . ., k-1 , if $Z \underset{T_i}{\longrightarrow} \alpha$ and $Z \underset{T_j}{\longrightarrow} \beta$ then α and β have the same set of big letters (and in fact none of them except for Z is recursive).

IV. TWO USEFUL RESULTS

In this section we present two results which are used later on in the proof of our main result.

Throughout this paper we shall often use phrases like "(sufficiently) long word x with a property P" or a "(sufficiently) long (sub)derivation with a property P". This will have the following meaning.

1) By a "(sufficiently) long word x with a property P" we mean a word x with property P which is longer than some constant C the computation of which does not depend on x itself.

2) By a "(sufficiently) long (sub)derivation with a property P" we mean a (sub)derivation D satisfying P of a word x which is longer than $|x|^C$ where C is a constant independent of either x or D.

The following result will be used quite often to get long subderivations from other long subderivations. Before we formulate it we need another definition.

Definition 11. Let f be a function from $\mathcal{R}_{pos}$ into $\mathcal{R}_{pos}$. We say that f is slow if

$$(\forall \alpha)_{\mathcal{R}_{pos}} (\exists n_\alpha)_{\mathcal{R}_{pos}} (\forall x)_{\mathcal{R}_{pos}} [\text{if } x > n_\alpha \text{ then } f(x) < x^\alpha].$$

Thus a constant function, $(\log x)^k$ and $(\log x)^{\log \log x}$ are examples of slow functions, whereas $(\log x)^{\log x}$, x^2, $\sqrt{x}$ are examples of functions which are not slow.

Let G be an EDTOL system and let g be a slow function. Let $\overline{D}$ be a long subderivation of a derivation D of x in G. Let us divide the words in $\overline{D}$ into classes in such a way that a number of classes is not larger than $g(|x|)$.

Lemma 1. There exists a long subderivation of D consisting of all the words which belong to one class of the above division into classes.

Proof.

As $\overline{D}$ is a long subderivation, it is longer than $|x|^C$ for some constant C (independent of $\overline{D}$ and x). Thus in our division there must be a class (say F) consisting of at least $\frac{|x|^C}{g(|x|)}$ elements. But $\frac{|x|^C}{g(|x|)} = \frac{|x|^{C-\alpha} \cdot |x|^{\alpha}}{g(|x|)}$ for every α. However for sufficiently long words x we have $\frac{|x|^{\alpha}}{g(|x|)} \geq 1$ and consequently $\frac{|x|^C}{g(|x|)} \geq |x|^{C-\alpha}$. Thus if we choose a subderivation in such a way that it consists of all words in F, then it is a long subderivation. Hence Lemma 1 is proved.

Now we shall present another result of a graph-theoretical nature which turns out to be very useful for our investigations.

First we need some definitions.

Definition 12. A $(n \times k)$ matrix of trees (abbreviated as $(n \times k)$ t-matrix) is a directed graph whose nodes form a $(n \times k)$ matrix which satisfies two conditions: (i) each node in the graph has at most one ancestor, (ii) if there is an edge leading from node (i, j) to node $(\overline{i}, \overline{j})$, for some $1 \leq i, i \leq n$ and $1 \leq j, j \leq k$, then $\overline{i} = i + 1$.

Definition 13. Let G_1 be a $(n \times k)$ t-matrix and let G_2 be a $(m \times k)$ t-matrix for some $m \leq n$. We say that G_2 is a sub-t-matrix of G_1 if the $(m \times k)$ matrix of nodes of G_2 is obtained by skipping some (may be none) rows from the matrix of nodes of G_1 and there is an edge between two nodes in G_2 if and only if this edge is in the transitive closure of G_1.

Definition 14. A $(n \times k)$ t-matrix G is said to be well formed if it satisfies the following two conditions:

1) If a node (i, j) has descendants then $(i + 1, j)$ is one of them.

2) If there is an edge leading from (i, j) to $(i + 1, \ell)$, then, for every p in $\{1, \ldots, n - 1\}$, there is an edge leading from (p, j) to $(p + 1, \ell)$.

The following result is proved in Ehrenfeucht and Rozenberg [3].

<u>Theorem 1</u>. For every positive integer k there exist positive reals r_k and s_k such that for every positive integer n and for every $(n \times k)$ t-matrix G there exists a well formed $(m \times k)$ t-matrix H which is a sub-t-matrix of G and for which $m \geq r_k n^{s_k}$.

V. ON THE EXISTENCE OF "LONG" NEAT SUBDERIVATIONS

In this section we prove the main result of this section.

First we need a definition.

Definition 15. Let Σ be a finite alphabet and let f be a function from $\mathbb{R}_{pos}$ into $\mathbb{R}_{pos}$. Let w be in Σ^*. We say that w is a f-random word (over Σ) if

$$(\forall w_1, u_1, w_2, u_2, w_3)_{\Sigma^*} \text{ [if } w = w_1u_1w_2u_2w_3 \text{ and } |u_1| > f(|w|) \text{ and } |u_2| > f(|w|), \text{ then } u_1 \neq u_2]$$

Thus, informally speaking, we call a word w f-random if every two disjoint subwords of w which are longer than $f(|w|)$ are different.

Theorem 2. For every EPDTOL system G and every slow function f there exist r in $\mathcal{R}_{pos}$ and s in N such that, for every w in L(G), if $|w| > s$ and w is f-random, then every derivation of w in G contains a neat subderivation longer than $|w|^r$.

Proof.

Let $G = \langle V, P, \omega, \Sigma \rangle$ be an EPDTOL system and let f be a slow function.

We shall present our proof in the form of a construction that takes several steps.

Before we start it, it is instructive to notice that if we consider an arbitrary derivation D of a f-random word then in each word of D each big letter has a unique occurrence.

STEP 1

Let x be a sufficiently large f-random word in L(G) and let D be a derivation of x in G. (It is clear that for sufficiently large x, ω must contain at least one big letter.) Now we choose a subderivation $D_1 = ((x_0^{(1)}, \cdots, x_{k_1}^{(1)}), (T_0^{(1)}, \cdots, T_{k_1-1}^{(1)}), \mathcal{O}_1)$ in the following fashion:

1) Let $x_0^{(1)} = \omega$, and

2) for a given $x_i^{(1)}$ we choose $x_{i+1}^{(1)}$ to be the nearest word in the derivation D such that it contains a big letter and it contains some small letters contributed from a big letter in $x_i^{(1)}$,

3) we continue to choose next elements $(x_{i+1}^{(1)})$ as long as possible.

Observation:

Let us note that D_1 is sufficiently long.

This is shown as follows:

Let $\# V = m_0$ and let m_1 be the maximal length of the right hand side of any production in any table from P. Let for i in $\{1, \cdots, k_1\}$, Δ_i denote the total number of occurrences contributed in D from all these small occurrences in $x_i^{(1)}$ which are contributed from big occurrences in $x_{i-1}^{(1)}$. Let Δ_ω denote the total number of occurrences contributed in D from all small occurrences in ω. Let Δ_{fin} denote the total number of occurrences contributed in D from all big occurrences in $x_{k_1}^{(1)}$.

It is clear that:

$\Delta_{fin} \leq m_0 \cdot m_1 \cdot f(|x|)$,

$\Delta_\omega < |\omega| \cdot f(|x|)$, and

for every i in $\{1, \ldots, k_1\}$, $\Delta_i \leq m_0 \cdot m_1 \cdot f(|x|)$.

But $|x| = \Delta_{fin} + \Delta_\omega + \sum_{i=1}^{k_1} \Delta_i$, hence

$|x| < f(|x|) \; (m_0 \cdot m_1 + |\omega| + m_0 \cdot m_1 \cdot k_1)$ and so

$$k_1 > \frac{\frac{|x|}{f(|x|)} - (m_0 \cdot m_1 + |\omega|)}{m_0 \cdot m_1}$$

As f is slow, for every ε in $\mathcal{R}_{pos}$ we can choose $|x|$ large enough so that $f(|x|) < |x|^\varepsilon$. Consequently (because m, q and $|\omega|$ are constants) for every α in $\mathcal{R}_{pos}$ we can adjust x in such a way that $k_1 > |x|^{1-\alpha}$.

STEP 2

Let us consider the derivation D_1 obtained in STEP 1.

Let us divide words in D_1 into classes in such a way that two words belong to the same class if and only if they contain the same set of big letters. As the number of such classes is clearly bounded by a constant, we can apply Lemma 1 and obtain a sufficiently long subderivation D_{11} of D.

Then let us divide words in D_{11} into classes in such a way that two words belong to the same class if and only if they contain the same set of big letters. Again applying Lemma 1 we obtain a sufficiently long subderivation D_{12} of D.

Then let us divide words in D_{12} into classes in such a way that two words belong to the same class if and only if they contain the same set of unique letters. Applying Lemma 1 we get in this way a sufficiently long D_{13} of D.

Then let us divide words in D_{13} into classes in such a way that two words belong to the same class if and only if they contain the same set of multiple letters. Applying Lemma 1 we get a subderivation D_{14} of D which is sufficiently large.

Finally let us divide words in D_{14} into classes in such a way that two words belong to the same class if and only if they contain the same set of letters. Again, applying Lemma 1 we get a sufficiently large subderivation of D. Let us call it D_2.

STEP 3

Let us consider the derivation $D_2 = ((x_0^{(2)}, x_1^{(2)}, \ldots, x_{k_2}^{(2)}), (T_0^{(2)}, \ldots, T_{k_2-1}^{(2)}), \mathcal{G}_2)$ obtained in STEP 2.

Let M_{D_2} be a t-matrix constructed as follows:

1) Every column in M_{D_2} corresponds to exactly one big letter in D_2 (their order is fixed, but arbitrary).

2) For every word in D_2 we have exactly one row in M_{D_2} (where for consecutive words in D_2 we have consecutive rows in M_{D_2}).

3) There is an edge leading from (i, j) to (i + 1, t) if and only if the t^{th} occurrence in $x_{i+1}^{(2)}$ is derived from the j^{th} occurrence in $x_i^{(2)}$.

As each occurrence in a derivation has only one ancestor and as each big occurrence in D_2 is unique, M_{D_2} is indeed a t-matrix. But then applying Theorem 1 we obtain a sufficiently long subderivation D_3 of D which corresponds to a well-formed sub-t-matrix of M_{D_2}.

Observation:

Note that directly from the fact that D_3 corresponds to a well-formed sub-t-matrix we have that:

1) In each word of D_3 there is a big recursive letter.

2) A letter is big recursive (big nonrecursive) in a word of D_3 if and only if it is big recursive in every word of D_3.

STEP 4

Let us consider the subderivation D_3 obtained in STEP 3.

Let us divide words in D_3 into classes in such a way that two words belong to the same class if and only if each small occurrence of the same letter in these words contributes the same number of occurrences in D. Clearly the total number of different classes obtained in this way does not exceed $f(|x|)^{m_0}$ (recall that $m_0 = \#V$). As f(n) is a slow function then so is $(f(n))^{m_0}$. Hence we can apply Lemma 1 to obtain a sufficiently long subderivation D_4 of D.

STEP 5

Let us consider the subderivation $D_4 = ((x_0^{(4)}, \ldots, x_{k_4}^{(4)}), (T_0^{(4)}, \ldots, T_{k_4}^{(4)}), \mathcal{O}_4)$ obtained in STEP 4.

Let us divide all small letters into classes in such a way that two small letters belong to the same class if they contribute the same number of occurrences in D. Each such class can be identified by the number of occurrences contributed to D by an element of this class. Starting with the class corresponding to the highest such number and then going through all "smaller" classes perform (one by one) the following.

For the highest number h_{max} construct a t-matrix $M_{D_4, h_{max}}$ in the following way:

1) Each column in $M_{D_4, h_{max}}$ corresponds to exactly one small letter from the class h in D_4 (their order is fixed, but arbitrary).

2) For every word in D_4 we have exactly one row in $M_{D_4, h_{max}}$ (where for consecutive words in D_4 we have consecutive rows in $M_{D_4, h_{max}}$).

3) There is an edge leading from (i + 1, j) to (i, t) if and only if the j^{th} occurrence in $x_{i+1}^{(4)}$ is derived from the t^{th} occurrence in $x_i^{(4)}$.

4) Turn the resulting graph upside down to obtain a "normal" t-matrix.

It is easy to check that we have really got a t-matrix and so we can apply Theorem 1 to obtain a sufficiently long subderivation of D.

Then from this subderivation we obtain a sufficiently long subderivation by exactly the same method but using the next to the highest (h_{max}) class. And so on until we exhaust all classes. Let us denote the resulting (sufficiently long) subderivation as D_5.

Observation:

It should be clear that if A is a small letter in A and it belongs to class h then in a direct derivation step in D_4 it either derives only itself (production $A \to A$) or it derives a string consisting of small letters from classes lower than h (production $A \to B_1 \dots B_q$ with $q \geqslant 2$ and each B_i, $1 \leqslant i \leqslant q$, in a class lower than this of A).

STEP 6

From the subderivation $D_5 = ((x_0^{(5)}, \dots, x_{k_5}^{(5)}), (T_0^{(5)}, \dots, T_{k_5}^{(5)}), \mathcal{P}_5)$ let us construct a subderivation D_6 by taking (starting from the top) all words $x_j^{(5)}$ where $j = m_0 \cdot u$ for $u \geqslant 0$ (so we get the sequence of words $x_0^{(5)}$, $x_{m_0}^{(5)}$, $x_{2m_0}^{(5)}$, ...).

Observation:

Clearly, D_6 is sufficiently long.

Now it must be that in D_6 in a direct derivation step a nonrecursive letter is rewritten as a string of small recursive letters, and a small recursive letter (already in D_5) must be rewritten as itself only.

This ends the construction.

To conclude the proof it is enough to observe that D_6 is indeed a neat subderivation. In fact we can note that:

(i) The condition 1 of Definition 10 was satisfied already in D_2.

(ii) The condition 2 of Definition 10 was satisfied already in D_5.

(iii) The condition 3 of Definition 10 was satisfied already in D_3.

(iv) The condition 4 of Definition 10 was satisfied already in D.

(v.1) The condition 5.1 of Definition 10 was satisfied already in D_1.

(v.2) The condition 5.2 of Definition 10 was satisfied only (in general) in D_6.

(vi) The condition 6 of Definition 10 was satisfied already in D_4.

(vii) The condition 7 of Definition 10 was satisfied already in D_3.

VI. DISCUSSION

First of all let us point out that restricting ourselves in Theorem 2 to f-random words only still leaves us (in general) with a considerable number of words providing that f is not "too slow". This is shown as follows.

Theorem 3. Let Σ be a finite alphabet such that $\#\Sigma = m \geq 2$. Let f be a function from $\mathcal{R}_{pos}$ into $\mathcal{R}_{pos}$ such that, for every x in $\mathcal{R}_{pos}$, $f(x) \geq 4 \log_2 x$. Then, for every positive integer n,

$$\frac{\#\{w \in \Sigma^* : |w| = n \text{ and } w \text{ is f-random}\}}{m^n} \geq 1 - \frac{1}{n} .$$

Proof.

Let Σ and f satisfy the statement of the theorem.

First let us find an upper bound on the number of words in Σ^* of length n which are not f-random.

1) If a word w is not f-random, then it can be written in the form

$a_1 \cdot \cdot \cdot a_{n_1} \alpha \; a_{n_1+|\alpha|+1} \cdot \cdot \cdot a_{n_2} \; \alpha \; a_{n_2+|\alpha|+1} \cdot \cdot \cdot a_n$ for some

$a_1, \ldots, a_{n_1}, a_{n_1+|\alpha|+1}, \ldots, a_n$ in Σ and α in Σ^+ where $|\alpha| > f(n)$.

2) With the fixed values of n_1, n_2 and $|\alpha|$ we may have at most $m^{|\alpha|} \cdot m^{n-2|\alpha|} = m^{n-|\alpha|}$ words which are not f-random. But $|\alpha| > f(n)$ and so $m^{n-|\alpha|} < m^{n-f(n)}$.

3) The number of choices for n_1, n_2 and α is not larger than n^3.

4) Thus the number of words of length n which are not f-random is smaller than $n^3 \cdot m^{n-f(n)}$.

Consequently,

$$\frac{\#\{w \in \Sigma^* : |w| = n \text{ and } w \text{ is not f-random}\}}{m^n} < \frac{n^3 \cdot m^{n-f(n)}}{m^n} = \frac{n^3}{m^{f(n)}} .$$

But $f(n) \geq 4 \log_2 n$ and so

$$\frac{n^3}{m^{f(n)}} \leq \frac{n^3}{m^{4 \log_2 n}} = \frac{n^3}{2^{\log_2 m \cdot 4 \log_2 n}} = \frac{n^3}{2^{\log_2 n \cdot 4 \log_2 m}} = \frac{n^3}{n^{4 \log_2 m}} \leq \frac{1}{n}$$

Thus

$$\frac{\#\{w \in \Sigma^*: |w| = n \text{ and } w \text{ is f-random}\}}{m^n} \geqslant 1 - \frac{1}{n}$$

which proves the theorem.

Hence, in general, it would be rather a problem to construct (EDTOL) languages which would not consist "mostly" of f-random words providing that f is not too slow ($f(x) \geq 4 \log x$).

Now let us notice that our notion of a neat subderivation and hence Theorem 2 were formulated for propagating EDTOL systems. It should be obvious to the reader that the notion of a neat subderivation can be reformulated for the case of a general EDTOL system (one would just allow "nonproductive letters" and then ignore them in a sense that all conditions from Definition 10 would be reformulated in such a way that nonproductive letters could appear "anywhere" in words of a subderivation and in productions of its tables). Then, the accordingly modified Theorem 2 obviously holds. In this sense the structure of a derivation in EPDTOL systems as presented so far, forms a "backbone" of the structure of a derivation in an arbitrary EDTOL system.

Also, the structural theorems of the type of Theorem 2 are mostly useful for inferring properties of (EDTOL) languages. Examples of such an approach are presented in Ehrenfeucht and Rozenberg [4] and in Ehrenfeucht and Rozenberg [5]. From this point of view it suffices to consider EPDTOL systems only because we have the following result.

Theorem 4. A language K is an EDTOL language if and only if $K - \{\Lambda\}$ is an EPDTOL language.

We leave the proof of this result to the reader because it can be proved by a direct construction following the idea from the proof of Theorem 5 in Rozenberg [7].

VII. REFERENCES

1. P.A. Christensen, Hyper AFL's and ETOL systems, in [9], 1974.

2. P. Downey, Developmental systems and recursion schemes, Proceedings of the Conference on Biologically Motivated Automata Theory, McLean, Virginia, 1974.

3. A. Ehrenfeucht and G. Rozenberg, On matrices of trees, Department of Computer Science, University of Colorado, Boulder, Technical Report Number CU-CS-044-74, 1974.

4. A. Ehrenfeucht and G. Rozenberg, On some context-free languages that are not deterministic ETOL languages, Department of Computer Science, University of Colorado, Boulder, Technical Report Number CU-CS-048-74, 1974.

5. A. Ehrenfeucht and G. Rozenberg, A pumping theorem for deterministic ETOL languages, Department of Computer Science, University of Colorado, Boulder, Technical Report Number CU-CS-047-74, 1974.

6. G.T. Herman and G. Rozenberg, Developmental systems and languages, North Holland Publishing Company, 1974.

7. G. Rozenberg, TOL systems and languages, Information and Control, v. 23, 357-381.

8. G. Rozenberg, Extension of tabled OL systems and languages, International Journal of Computer and Information Sciences, v. 2, 311-334.

9. G. Rozenberg and A. Salomaa, L systems, Lecture Notes in Computer Science, v. 15, Springer-Verlag, 1974.

10. A. Salomaa, Recent results on L systems, Proceedings of the Conference on Biologically Motivated Automata Theory, McLean, Virginia, 1974.