

ON AMBIGUITY IN EOL SYSTEMS

by

A. Ehrenfeucht*

and

G. Rozenberg**

#CU-CS-158-79

August, 1979

*A. Ehrenfeucht, Department of Computer Science, University of
Colorado at Boulder, 80309 USA

**G. Rozenberg, Institute of Applied Mathematics and Computer
Science, University of Leiden, 2300 RA Leiden, Holland

All correspondence to G. Rozenberg

ON AMBIGUITY IN EOL SYSTEMS

by

A. Ehrenfeucht
Department of Computer Science
University of Colorado at Boulder
Boulder, Colorado 80309
U.S.A.

and

G. Rozenberg
Institute of Applied Mathematics and Computer Science
University of Leiden
2300 RA Leiden
Holland

All correspondence to G. Rozenberg

ON AMBIGUITY IN EOL SYSTEMS

by

A. Ehrenfeucht
Department of Computer Science
University of Colorado at Boulder
Boulder, Colorado 80309
U.S.A.

and

G. Rozenberg
Institute of Applied Mathematics and Computer Science
University of Leiden
2300 RA Leiden
Holland

All correspondence to G. Rozenberg

ABSTRACT

It is demonstrated that the degree of ambiguity of a context free language K in the class of EOL systems is not larger than the degree of ambiguity of K in the class of context free grammars. A language K is said to have the negative prefix property if no word K catenated with a nonempty prefix of a word from K^+ yields an element of K . It is shown that if an EOL language has the negative prefix property and is EOL-unambiguous then K^+ is also EOL-unambiguous. Using those results several conjectures concerning ambiguity of EOL languages are disproved.

INTRODUCTION

The topic of ambiguity of context free grammars and languages is one of the classical topics of formal language theory (see, e.g., [P] and [S]). The class of EOL systems (see, e.g., [RS]) forms a very natural extension of the class of context free grammars.

Investigating the ambiguity of EOL systems and languages forms then a "natural extension" of the research on ambiguity of context free grammars and languages. However very little is known on this topic: [ReS] treats some aspects of ambiguity of OL systems and languages and in [MSW] some observations are made on the ambiguity of EOL systems and languages concerning mainly EOL forms. In particular such a research is needed to put the relationship between the class of EOL systems (languages) and the class of CF grammars (languages) in proper perspective because simple intuition leads one to a conclusion that while a context free language is defined in the class of EOL systems its degree of ambiguity might be spoiled; in particular an unambiguous context free language may be inherently ambiguous EOL language. This is conjectured in [MSW].

In this paper we begin a systematic study of ambiguity of EOL systems and languages and in particular we disprove the above conjecture. That is, we show that the EOL-ambiguity of a context free language cannot be larger than its context free ambiguity. Since it was observed already in [MSW] that there exist inherently ambiguous context free languages that are EOL-unambiguous we get in this way (yet another) point in favor of EOL systems and languages.

We also provide a condition under which an EOL language K^+ is EOL-unambiguous if K is EOL-unambiguous. This allows us to disprove some other conjectures concerning ambiguity of EOL languages.

PRELIMINARIES

We assume the reader to be familiar with the basics of the theory of EOL systems and languages and the basics of the theory of context free languages (see, e.g., [RS] and [S]). To settle the notation for this paper we recall the definition of an EOL system.

Definition. An EOL system is a construct $G = (\Sigma, h, S, \Delta)$ where Σ is a finite alphabet, h is a finite substitution from Σ into the power set of Σ^* (extended homomorphically to Σ^*), $S \in \Sigma \setminus \Delta$, S is called the axiom of G , and $\Delta \stackrel{\subset}{\neq} \Sigma$, Δ is called the terminal alphabet of G . The language of G is defined by $L(G) = \{x \in \Delta \mid x \in h^n(S) \text{ for some } n \geq 1\}$. \square

Mostly in formal language theory one considers two languages equal if they differ at most by the empty word. For this reason we assume that no languages we consider contain the empty word.

Usually, the definition of the (degree of) ambiguity of a grammar is based on counting the number of distinct derivation trees for a word in the grammar. We will consider also an alternative way of defining ambiguity which is based on counting the number of prime (derivation) trees defined as follows.

Definition. A labelled tree T is called prime if there exists a path τ in T leading from the root of T to a leaf of T with the following property: if v is a node on τ such that v has exactly one direct descendant, then the label of v is different from the label of its descendant. \square

Definition. Let G be an EOL system.

(1). We say that G is ambiguous of degree k , where k is a positive

integer, if every word in $L(G)$ possesses at most k distinct derivation trees in G and, moreover, some word in $L(G)$ possesses exactly k distinct derivation trees. G is ambiguous of degree ∞ if, for any positive integer k , there exists a word in $L(G)$ possessing more than k distinct derivation trees. If G is ambiguous of degree 1 then we say that G is unambiguous. We use amb G to denote the degree of ambiguity of G .

(2). We say that G is p-ambiguous of degree k , where k is a positive integer, if every word in $L(G)$ possesses at most k distinct prime derivation trees in G and, moreover, some word in $L(G)$ possesses exactly k distinct prime derivation trees. G is p-ambiguous of degree ∞ if, for any positive integer k , there exists a word in $L(G)$ possessing more than k distinct prime derivation trees. If G is p-ambiguous of degree 1 then we say that G is p-unambiguous. We use pamb B to denote the degree of p-ambiguity of G . \square

We end this section by demonstrating that p-ambiguity is an "honest" measure of ambiguity of an EOL system in a sense that in every EOL system G every word in $L(G)$ has at least one prime derivation tree. Thus $L(G)$ remains unaltered even if only prime derivation trees are taken into account.

Theorem 1. Let G be an EOL system. Every word in $L(G)$ has a prime derivation tree.

Proof.

Let $w \in L(G)$ and let T be a derivation tree of w in G . If T is not prime then let $T^{(1)}$ be a derivation tree constructed from T as follows: on every path in T leading from the root to a leaf choose the earliest (the nearest to the root of T) occurrence of a node with one direct descendant only such that both the node and its

descendant have the same label; in each case replace the subtree rooted at the node by the subtree rooted at its descendant. Obviously, $T^{(1)}$ is a derivation tree of w in G . If $T^{(1)}$ is prime, then we have got a prime derivation of w in G . Otherwise we iterate the above construction obtaining in this way the sequence $\tau = T, T^{(1)}, \dots$ of derivation trees of w in G . Since T is a finite tree, τ is finite and its last element is a prime derivation tree of w in G .

Consequently each word in $L(G)$ has a prime derivation tree. \square

RESULTS

We start by comparing the ambiguity and the p-ambiguity of EOL systems and languages.

Lemma 1. (i). For every EOL system G , $\underline{\text{pamb}} G \leq \underline{\text{amb}} G$.

(ii). For every EOL language K , $\underline{\text{pamb}} K \leq \underline{\text{amb}} K$.

Proof

Obvious. \square

In proving various properties of EOL systems and languages one often transforms a given EOL system to obtain another one generating the same language, but "better" to deal with. The following result describes transformations which do not spoil the ambiguity (p-ambiguity) of the system considered.

Lemma 2. Let G and H be EOL systems, such that $L(G) = L(H)$.

If there exists a function ϕ mapping every successful derivation tree in G into a derivation tree of the same word in H such that:

- (1). the range of ϕ is the set of all successful derivation trees in H , and
 - (2). for a successful derivation tree T , if $\phi(T)$ is prime then T is prime,
- then $\underline{\text{amb}} H \leq \underline{\text{amb}} G$ and $\underline{\text{pamb}} H \leq \underline{\text{pamb}} G$.

Proof.

Obvious. \square

The following is a variation of a very useful normal form for EOL systems used often in the literature.

Definition. Let $G = (\Sigma, h, S, \Delta)$ be an EOL system. We say that G is in normal form if G is synchronized⁽¹⁾, $S \notin h(b)$ for every $b \in \Sigma$, and if $\alpha \in h(b)$ for $b \in \Sigma$, then either $\alpha \in \Delta^*$ or $\alpha \in (\Sigma \setminus (\Delta \cup \{F, S\}))^+$

or $\alpha = F$, where F is the synchronization symbol of G . \square

The following two lemmas establish the usefulness of EOL systems in normal form as far as ambiguity is concerned.

Lemma 3. For every EOL system G there exists an EOL system H such that $L(H) = L(G)$, H is in normal form, $\underline{\text{amb}} H \leq \underline{\text{amb}} G$ and $\underline{\text{pamb}} H \leq \underline{\text{pamb}} G$.

Proof.

This follows from an easy modification of a well known technique for constructing an EOL system H (for a given EOL system G) which is propagating, in normal form and such that $L(G) = L(H)$; see [RS], proofs of theorems If one skips the part of this construction yielding a propagating system, then one gets directly a function ϕ satisfying the statement of Lemma 2. Hence the result follows from Lemma 2. \square

One might be inclined to think that "stable" productions $A \rightarrow A$ are an additional source of ambiguity for EOL systems. However we shall show that this need not be the case. The main tool in this argument is the construction in the following lemma.

Lemma 4. Let G be an EOL system in normal form. There exists an EOL system H such that $L(G) = L(H)$ and $\underline{\text{amb}} H = \underline{\text{pamb}} H = \underline{\text{pamb}} G$.

Proof.

Let $G = (\Sigma, h, S, \Delta)$.

Let $H = (\bar{\Sigma}, \bar{h}, S, \Delta)$ be the EOL system constructed as follows.

(1). Let $W(G) = \Sigma \setminus (\Delta \cup \{F, S\})$, where F is the synchronization symbol of G and let $\bar{W}(G) = \{b_i \mid 1 \leq i \leq 3 \text{ and } b \in W(G)\}$. Then $\bar{\Sigma} = \bar{W}(G) \cup \Delta \cup \{F, S\}$.

(2). The finite substitution \bar{h} is determined as follows.

(2.1). Let $b \in W(G)$.

If $b \in h(b)$, then $b_3 \in \bar{h}(b_2)$ and $b_3 \in \bar{h}(b_3)$.

If $\alpha \in h(b)$ and $\alpha \in \Delta^*$, then $\alpha \in \bar{h}(b_1)$ and $\alpha \in \bar{h}(b_3)$.

If $\alpha \in h(b)$, $\alpha \neq b$ and $\alpha \in (W(G))^+$, then

$z \in \bar{h}(b_1)$ for every z in Z_α , where Z_α is the set of all words resulting from α by attaching to at least one (occurrence of a) letter in α the index 1 and attaching the index 2 to all remaining (occurrences of) letters, $\alpha_2 \in \bar{h}(b_2)$ where α_2 is the word resulting from α by attaching the index 2 to every (occurrence of a) letter in α , and $\alpha_3 \in \bar{h}(b_3)$, where α_3 is the word resulting from α by attaching the index 3 to every (occurrence of a) letter in α .

(2.2). If $\alpha \in h(S)$ and $\alpha \in \Delta^*$, then $\alpha \in \bar{h}(S)$.

If $\alpha \in h(S)$ and $\alpha \in (W(G))^+$ then $z \in \bar{h}(S)$ for every z in Z_α .

(2.3). For every $b \in \bar{\Sigma}$, $F \in \bar{h}(b)$.

(2.4). \bar{h} is completely determined by (2.1) through (2.3).

The following observations follow directly from the construction of H .

(i). Let \bar{T} be a derivation tree of a word x in $L(H)$. Let T be the tree resulting from \bar{T} by changing each label of a node in \bar{T} that is of the form b_i , $1 \leq i \leq 3$, $b \in W(G)$, into b . Then T is a derivation tree of x in $L(G)$; we say that T corresponds to \bar{T} .

(ii). Let \bar{T} be a derivation tree of a word in $L(H)$ and let v be an internal node of \bar{T} . Let T be the corresponding tree in G .

If the label of v is of the form b_1 , $b \in W(G)$, then on the path leading from the root of T to v there is no identity (that is there is no node e such that e has only one direct descendant and the labels of e and its direct descendant are identical) and there is a path leading from v to a leaf of T on which there is no identity. If the label of v is

of the form b_2 , $b \in W(G)$, then on the path leading from the root of T to v there is no identity but on every path leading from v to a leaf of T there is an identity.

If the label of v is of the form b_3 , $b \in W(G)$, then there is an identity on the path leading from the root of T to v .

(iii). Let T be a prime derivation tree of a word in $L(G)$. There is precisely one way of changing every label in T which is an element of $W(G)$ into a label from $\overline{W(G)}$ in such a way that the resulting tree \overline{T} is a derivation tree of a word in $L(H)$. On the other hand, if T is not prime, it is not at all possible to obtain from T a derivation tree \overline{T} of a word in $L(H)$.

(iv). Every derivation tree of a word in $L(H)$ is prime.

Now from (i) through (iv) it follows that $L(H) = L(G)$ and if $x \in L(H) = L(G)$, then:

the number of different prime derivation trees of x in $L(H)$
equals

the number of different derivation trees of x in $L(H)$
equals

the number of different prime derivation trees of x in $L(G)$.

Hence the lemma holds. \square

We can prove now that both notions of ambiguity of EOL languages that we consider are equivalent.

Theorem 2. For every EOL language K , $\underline{\text{amb}} K = \underline{\text{pamb}} K$.

Proof.

(i). From Lemma 1 we have $\underline{\text{pamb}} K \leq \underline{\text{amb}} K$.

(ii). Let G_0 be an EOL system defining the p -ambiguity of K , that is $L(G_0) = K$ and $\underline{\text{pamb}} G_0 = \underline{\text{pamb}} K$. By Lemma 3 we can assume that G_0 is in normal form. Hence by Lemma 4 there exists an EOL

system H such that $\text{amb } H = \text{pamb } H = \text{pamb } G_0$ and $L(H) = L(G_0)$. Consequently $\text{amb } K \leq \text{pamb } K$.

The theorem follows from (i) and (ii). \square

The above result allows us to show that if we consider a context free language in the family of EOL systems then we do not spoil its degree of ambiguity.

Theorem 3. Let K be a context free language such that the degree of context free ambiguity of K equals k (where k is either a positive integer or ∞). Then $\text{amb } K \leq k$.

Proof.

If $k = \infty$ then the result is obvious.

Let k be a positive integer and let G be a context free grammar such that $L(G) = K$ and the degree of ambiguity of G equals k . Clearly we can assume that G is reduced, and so G does not contain rules of the form $B \rightarrow B$, B a nonterminal of G . Now let us use the standard construction to obtain an EOL system H such that $L(H) = K$; that is we add to productions of G productions of the form $b \rightarrow b$, b a terminal symbol of G . Clearly $\text{pamb } H$ equals the degree of ambiguity of G . Hence the result follows from Theorem 2. \square

The reader should note that the above theorem cannot be strengthened into "if and only if" result for context free languages. It was pointed out in [MSW] that the language $K_0 = \{a^n b^n c^m \mid m, n \geq 1\} \cup \{a^m b^n c^n \mid m, n \geq 1\}$ which is well known to be inherently ambiguous context free language (see [P]); is EOL-unambiguous.

Thus altogether we arrived at an important feature of the class

of EOL systems. They not only define a class of languages larger than the class of context free languages but they never spoil the context free degree of ambiguity of a language and sometimes they can improve the degree of ambiguity. In particular CF-unambiguous languages remain EOL-unambiguous while some CF-ambiguous languages become EOL-unambiguous.

Let $K_1 = \{a^i b^i \mid i \geq 1\}$. It is conjectured in [MSW] that $\text{amb } K_1^+ = \infty$. We disprove this conjecture now: it turns out that K_1^+ is EOL-unambiguous!!!

Corollary 1. K_1^+ is an EOL-unambiguous language.

Proof.

This follows from Theorem 3 and from an easy observation that $\{a^i b^i \mid i \geq 1\}^+$ is a CF-unambiguous language. \square

Now we turn to a result that will allow us to prove (in some cases) that K^+ remains EOL-unambiguous if K is EOL-unambiguous.

First, we need the following definition.

Definition. A language K has a negative prefix property if for all $x \in K$, $z \in K^+$ the following holds: if y is a nonempty prefix of z then $xy \notin K$.

Theorem 4. Let K be a language with a negative prefix property. If K is EOL-unambiguous then so is K^+ .

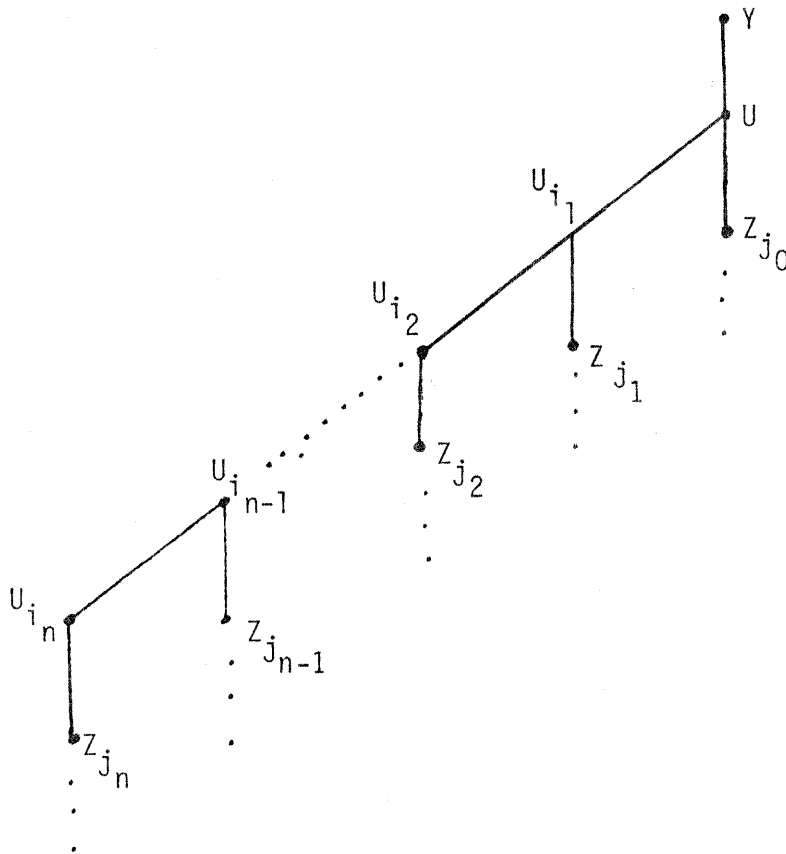
Proof.

Let $\bar{G} = (\Sigma, \bar{h}, S, \Delta)$ be an unambiguous EOL system generating K . By Lemma 3 we can assume that \bar{G} is in normal form and by Lemma 4 we can assume that each successful derivation tree in \bar{G} is prime.

Let U, Z, Y be new symbols, $U, Z, Y \notin \Sigma$, and let $G = (\Sigma \cup \{U, Z, Y\}, h, Y, \Delta)$ be the EOL system where h results by extending \bar{h} to $\Sigma \cup \{U, Z, Y\}$ as follows:
 $h(Y) = \{U\}$, $h(U) = \{UZ, Z\}$ and $h(Z) = \{Z, S\}$. Then let H result from G by the construction from the proof of Lemma 4.

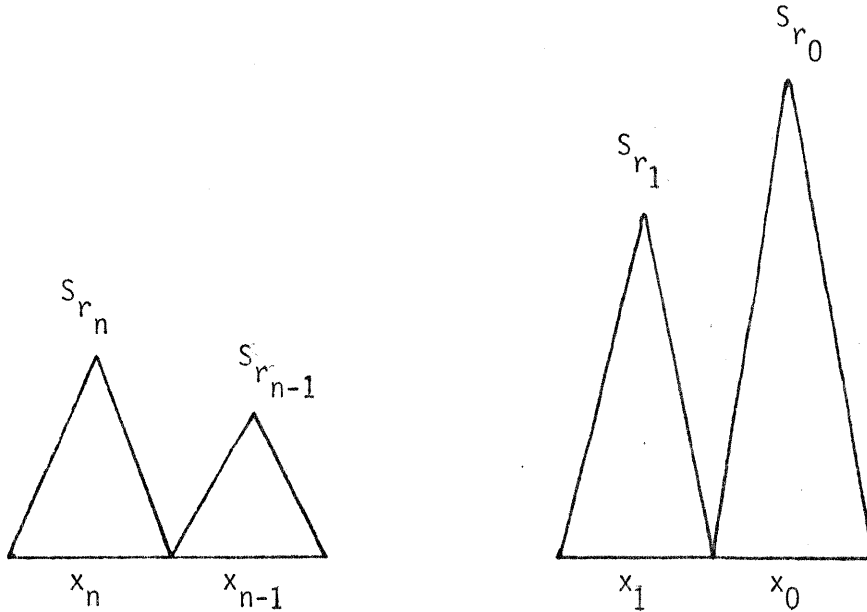
First of all we notice that $L(H) = K^+$ and every successful derivation tree in H is prime.

Moreover for each word in $L(H)$ there is only one prime derivation tree; this follows from the obvious fact that there is exactly one way of "gluing" a tree of the form $(n \geq 1, i_1, \dots, i_n, j_0, \dots, j_n \in \{1, 2\})$:



with a forest of the form (resulting from "concatenation" in H trees corresponding to successful derivation trees from G;

$r_0, \dots, r_n \in \{1, 2, 3\}$, $x_0, \dots, x_n \in L(G)$):



to obtain a (prime) successful derivation tree in H.

The fact that K has a negative prefix property ensures that this is the only way to obtain a derivation tree for $x_n \dots x_0$ in H.

Consequently the theorem holds. \square

Let $K_2 = \{a^i b^2 \mid i \geq 1\}$. It is conjectured in [MSW] that $\text{amb } K_0^+ = \text{amb } K_2^+ = \infty$. We disprove this conjecture by showing that $\text{amb } K_0^+ = \text{amb } K_2^+ = 1$, where K_0 is as defined above.

Corollary 2. Both K_0^+ and K_2^+ are EOL-unambiguous.

Proof.

Obviously both K_0 and K_2 possess negative prefix property. It is obvious that K_2 is EOL-unambiguous and it is shown in [MSW] that K_0 is EOL-unambiguous. Hence the result follows from Theorem 4. \square

FOOTNOTES

(1). For the purpose of this paper we assume that an EOL system $G = (\Sigma, h, S, \Delta)$ is synchronized if for every terminal symbol b if $x \in h(B)$ then either $x = \Lambda$ or $x = F$, where F is the synchronization symbol of G .

ACKNOWLEDGMENTS

The authors are indebted to R. Verraedt and especially to A. Salomaa for useful comments on the first draft of this paper. They gratefully acknowledge the financial support of NSF grant number MCS79-03838.

REFERENCES

- [P] Parikh, R., 1966, On context free languages, Journal of the A.C.M., 13, 510-581.
- [MSW] Maurer, H.A., Salomaa, A. and Wood, D., 1979, Synchronized EOL forms, Technical Rep. No. 79-CS-2, Dept. of Applied Math., McMaster University, Hamilton, Ontario.
- [ReS] Reedy, A. and Savitch, W., 1975, Ambiguity in the developmental systems of Lindenmayer, Journal of Comp. and Syst. Sc., 11, 262-283.
- [RS] Rozenberg, G. and Salomaa, A., 1979, The mathematical theory of L systems, Academic Press, New York - London, to appear.
- [S] Salomaa, A., 1973, Formal languages, Academic Press, New York - London.