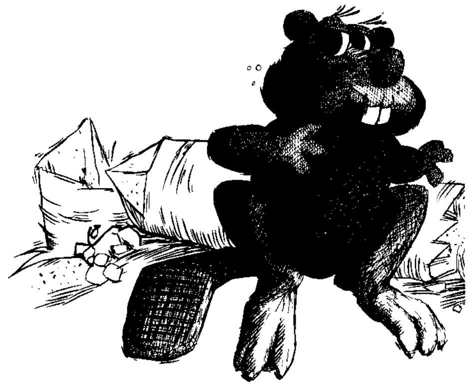
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT



Structural Equivalence of E0L Grammars

Thomas Ottmann
and
Derick Wood

Data Structuring Group
Research Report
CS-89-40

September, 1989

Structural Equivalence of E0L Grammars *

Thomas Ottmann[†] Derick Wood[‡]

Abstract

We investigate the structural equivalence decision problem for E0L grammars. While we are unable to solve it in its full generality, we solve it for some restricted classes of grammars. Second, we establish some reduction results that preserve structural equivalence that in the context-free case are sufficient to solve the problem, but unfortunately not in the E0L case. Third and finally, we illustrate with examples why the problem is much more difficult for E0L grammars than it is for context-free grammars.

1 Introduction

Context-free grammars and E0L grammars are two popular means of generating languages by iterative rewriting. In both cases the set of symbols is divided into terminal and nonterminal symbols and words consisting of only terminal symbols belong to the language generated by a grammar. The major difference between the two generating devices is that rewriting is carried out in *parallel* (and synchronously) for E0L grammars and *sequentially* (and asynchronously) for context-free grammars. Two grammars are *equivalent* if they generate the same language. The (language) equivalence problem is undecidable for context-free and for E0L grammars. Even very restricted cases of this problem turn out to be undecidable.

To each derivation in a context-free and E0L grammar we can associate a syntax tree. It contains the sentence symbol at the root and each branching node is labeled with a terminal or nonterminal symbol. The labeling is determined by the productions used in the derivation. Syntax trees of E0L grammars are labeled trees of uniform depth; syntax trees of context-free

*This work was supported under a Natural Sciences and Engineering Research Council of Canada Grant No. A-5692 and under a grant from the Information Technology Research Centre.

[†]Institut für Informatik, Universität Freiburg, Rheinstraße 10-12, D-7800 Freiburg, West Germany.

[‡]Data Structuring Group, Department of Computer Science, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada.

grammars can have arbitrary shapes. Considering syntax trees leads to the notion of *structural equivalence* between grammars. Two structurally equivalent grammars not only generate the same terminal words (or language) but also structure these words in the same manner. It has been shown by McNaughton [1] in 1967 that structural equivalence is decidable for context-free grammars. Surprisingly, the structural equivalence problem for E0L grammars has not attracted much interest thus far. It is the purpose of this paper to draw the reader's attention to the structural equivalence problem for E0L grammars, to illustrate a number of differences between the case of context-free grammars and the case of E0L grammars, to solve some special cases of the structural equivalence problem for E0L grammars, and to prove some reduction results for E0L grammars which preserve structural equivalence.

As we will see, parallel rewriting makes structural equivalence a challenging basic problem for E0L grammars which is considerably different from the corresponding problem for context-free grammars.

We assume that the reader is familiar with the basic notions for context-free grammars; for example, see [4] or [5]. In Section 2 we provide the necessary definitions for E0L grammars and some special cases of the structural equivalence problem are solved in Section 3. Section 4 introduces the notion of context equivalence and shows how to effectively reduce an E0L grammar with respect to this equivalence relation and Section 5 provides some concluding remarks.

2 Definitions and examples

An *E0L grammar* G is a quadruple (N, Σ, P, S), where N is an alphabet of *nonterminals*, Σ is an alphabet of *terminals*, $P \subseteq N \times (N^+ \cup \Sigma^+)$ is a finite set of *productions*, and $S \in N$ is the *sentence symbol.*

This definition differs from the classical one, see [3] for example, in two respects. First, only nonterminals have productions; the grammar is *synchronized.* Second, the right hand sides of productions cannot be the empty word; the grammar is *propagating.* These modifications do not affect the languages generated by E0L grammars (apart from the loss of the empty word). Throughout this paper we use upper case letters to denote nonterminals and lower case letters to denote terminals.

Rewriting is defined in the usual way. Let α be a nonempty word over N; that is, $\alpha = A_1 \dots A_n$, where A_i is in N, $1 \le i \le n$ and $n = |\alpha|$. α can be rewritten as $\beta = \beta_1 \dots \beta_n$, for some $\beta_i \in N^+ \cup \Sigma^+$, $1 \le i \le n$, if $A_i \to \beta_i$ is in P, $1 \le i \le n$. We usually denote this by $\alpha \Rightarrow \beta$. We write $\alpha \Rightarrow^d \beta$ to denote that α yields β in d steps, for $d \ge 0$, if: $d = 0$ and $\alpha = \beta$; $d = 1$ and $\alpha \Rightarrow \beta$; or $d > 1$ and there exists γ in N^+ such that $\alpha \Rightarrow \gamma$ and $\gamma \Rightarrow^{d-1} \beta$. We write

$\alpha \Rightarrow^+ \beta$ if $\alpha \Rightarrow^d \beta$, for some $d \geq 1$, and we write $\alpha \Rightarrow^* \beta$, if $\alpha \Rightarrow^d \beta$ for some $d \geq 0$. We say that $\alpha \Rightarrow^+ \beta$ and $\alpha \Rightarrow^* \beta$ are *derivations*. A derivation starting with the sentence symbol is called a *sentential derivation.*

Note that only purely nonterminal words can be rewritten. This is the reason for only allowing right hand sides of productions to be either completely nonterminal or completely terminal.

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

$$L(G) = \{x : x \in \Sigma^+ \text{ and } S \Rightarrow^+ x\}.$$

We say that an E0L grammar $G = (N, \Sigma, P, S)$ is *reduced* if the sentence symbol generates a terminal word and each nonterminal appears in at least one sentential derivation of a terminal word. From now on we tacitly assume that E0L grammars are reduced if not explicitly stated otherwise.

With each derivation $S \Rightarrow^+ \alpha$, $\alpha \in N^+ \cup \Sigma^+$, we can associate a *derivation tree* $tr(S \Rightarrow^+ \alpha)$. This is a uniform depth tree whose internal nodes are labeled with nonterminal symbols and whose external nodes are labeled with the symbols in α in left-to-right order. It also satisfies the following condition: For all internal nodes u, if u has r children $u_1, \ldots, u_r$, for some $r \geq 1$, then $L(u) \rightarrow L(u_1) \ldots L(u_r)$ is in P, where $L(v)$ denotes the label of node v.

A syntax tree whose root is labeled with the sentence symbol and whose leaves are labeled with terminal symbols is called a *sentential syntax tree.* Concatenating the labels at the frontier of a sentential syntax tree in left-to-right order gives a terminal word x generated by the grammar. Note that a derivation $S \Rightarrow^d x$ for $x \in \Sigma^+$, has a sentential syntax tree $tr(S \Rightarrow^d x)$ of height d.

Two E0L grammars G and G' are (language) *equivalent* if $L(G) = L(G')$. Two E0L grammars G and G' are *structurally equivalent* if they are equivalent and, moreover, their sentential syntax trees are structurally identical; that is, they differ only in the labels of their internal nodes. Because a terminal word generated by an E0L grammar may have more than one derivation the latter condition in the above definition of structural equivalence is to be understood as follows: For each sentential syntax tree T generating a terminal word $x \in L(G)$ there is a structurally identical syntax tree T' in G' generating the same word $x \in L(G')$, and vice versa. In Section 4 we will reduce structural equivalence to equivalence by the use of parenthesized versions of grammars.

An E0L grammar is *unambiguous* if each terminal word generated by the grammar has exactly one derivation tree. An E0L grammar $G = (N, \Sigma, P, S)$ is *invertible* if no two productions in P have the same right hand side. An invertible grammar may be ambiguous; however, there cannot be two different syntax trees with the same structure for the same terminal word if the grammar is invertible.

An E0L grammar $G = (N, \Sigma, P, S)$ is called *nonterminal complete*, or *n-complete*, for short, if for each nonterminal $X \in N$ there is at least one production $X \to \alpha$ in P, where $\alpha \in N^+$.

We will illustrate these notions by a number of examples. In each case the given E0L grammar has no useless nonterminals; that is, it is reduced, and the sentence symbol is always denoted by S.

Example 2.1 Let G_1 be given by

$$S \to AA,\ A \to AB,\ A \to a,\ B \to b.$$

Then, $L(G_1) = \{aa, abab\}$. The E0L grammar G_1 is invertible and unambiguous.

Example 2.2 Let G_2 be given by

$$S \to AS|a,\ A \to AA|a.$$

Then, $L(G_2) = \{a^{2^i} | i \geq 0\}$. G_2 is not invertible, because there are two productions with right hand side a. However, it can easily be seen that each $x \in L(G_2)$ has exactly one syntax tree. Hence, G_2 is unambiguous.

Example 2.3 The grammar G_3 defined by

$$S \to SS|a$$

is invertible, unambiguous, and structurally equivalent to G_2.

Example 2.4 The E0L grammar G_4 defined by

$$S \to AA|a,\ A \to S$$

is invertible and equivalent to both G_2 and G_3. However, G_4 is not structurally equivalent to G_2 or G_3.

Example 2.5 Let G_5 be given by the productions

$$S \to AA,\ A \to A|a.$$

Then, $L(G_5) = \{aa\}$. G_5 is invertible and ambiguous (there are infinitely many syntax trees for aa).

Note that the grammar G_1 is not n-complete, but $G_2, \ldots, G_5$ are n-complete.

Theorem 2.1 *Let $G = (N, \Sigma, P, S)$ be an E0L grammar. Then, an n-complete structurally equivalent E0L grammar $G' = (N', \Sigma, P', S)$ can be effectively constructed from G such that invertibility is maintained, that is, G' is invertible if G is invertible.*

Proof: N' and P' are obtained from N and P, respectively, by adding "looping" nonterminals and productions as follows. For each nonterminal $X \in N$ such that there is no production $X \to \alpha$ in P with $\alpha \in N^+$, choose a new nonterminal X', add X' to N' and the productions $X \to X'$, $X' \to X'X'$ to P. Obviously, G' is n-complete. G' is structurally equivalent to G because the derivations of terminal words are not affected by the new productions. Finally, all new productions have pairwise distinct right hand sides which are different from the right hand sides of the productions in P. Therefore, G' is invertible if G is invertible. Observe that G' is not reduced because the looping nonterminals do not occur in any sentential derivation of a terminal word. □

Invertibility allows us to reconstruct the labeling of a syntax tree of a given structure bottom up: If the labels of the leaves are given we can uniquely reconstruct the labels at all internal nodes. Requiring invertibility does not affect the languages generated by E0L grammars as the following theorem shows.

Theorem 2.2 *Let $G = (N, \Sigma, P, S)$ be an E0L grammar. Then, an invertible structurally equivalent E0L grammar $G' = (N', \Sigma, P', S')$ can be effectively constructed from G.*

Proof: This is a straightforward modification of the proof given in [2] for the case where $\Sigma = \{a\}$ and a set of sentence symbols instead of a single sentence symbol is allowed. Define N' to be the set $\{X \subseteq N : X \neq \emptyset\}$ and S' the set $\{X : X \subseteq N \text{ and } X \cap S \neq \emptyset\}$. Given a word α' over N' we say a word α over N *corresponds* to α' if $|\alpha| = |\alpha'|$ and each nonterminal symbol in α belongs to the set of nonterminal symbols appearing at the same position in α'.

The set P' of productions is defined as follows.

(i) P' contains a production $X \to \alpha'$, for $\alpha' \in (N')^+$, if and only if

$$X = \{A : A \in N, A \to \alpha \in P \text{ and } \alpha \text{ corresponds to } \alpha'\}$$

(ii) P' contains a production $X \to w$, for $w \in \Sigma^+$, if and only if

$$X = \{A : A \to w \in P\}.$$

It is not difficult to prove that G' is invertible and G and G' are structurally equivalent. □

Note that an E0L grammar with only one nonterminal is always invertible.

3 The one and two nonterminal cases

McNaughton's proof of the decidability of the structural equivalence problem for context-free grammars [1] is based on the following idea. For each context-free grammar a structurally equivalent and simplified grammar is constructed; a simplified grammar is obtained by appropriately identifying nonterminals which play the same role in all derivations. Simplifying drastically reduces the class of structurally equivalent context-free grammars; any two simplified structurally equivalent context-free grammars are isomorphic. This provides a basis for the decidability result. To test whether two context-free grammars are structurally equivalent we simplify them and test whether the resulting grammars are isomorphic.

An obvious approach to solving the structural equivalence problem for E0L grammars is to proceed in a similar manner. This suggests imposing additional requirements on E0L grammars which preserve structural equivalence but restrict the variety of grammars having these properties. Invertibility is an example of such a property. Restricting the number of nonterminals is another obvious choice. In this section we show that both restrictions can force an E0L grammar to be uniquely determined up to isomorphism.

Theorem 3.1 *Let $G = (N, \Sigma, P, S)$ be an E0L grammar with $N = \{S\}$. If $G' = (N', \Sigma, P', S')$ is structurally equivalent to G and invertible, then G and G' are isomorphic.*

Proof: Because G has only one nonterminal its productions must be of the following forms

$$S \rightarrow S^{i_1}|\cdots|S^{i_k},$$

where $i_j \geq 1$, for $1 \leq j \leq k$, and

$$S \rightarrow w_1|\cdots|w_l,$$

where $w_j \in \Sigma^+$, for $1 \leq j \leq l$. Clearly, G' must also have productions $S' \rightarrow w_j$, $1 \leq j \leq l$, and cannot have productions $A \rightarrow w_j$, where $A \in N'$ and $A \neq S'$, otherwise G' would not be invertible. Now consider the syntax tree of Figure 1. Because G and G' are structurally equivalent, there is a syntax tree of the same structure in G' which generates the same terminal word. This is only possible if $S' \rightarrow S'^{i_j}$ is in P'. Because of invertibility no other nonterminal can generate S'^{i_j}. Thus, the productions in P and P' are isomorphic. □

Theorem 3.1 can be extended to the case where both E0L grammars have exactly two nonterminals.

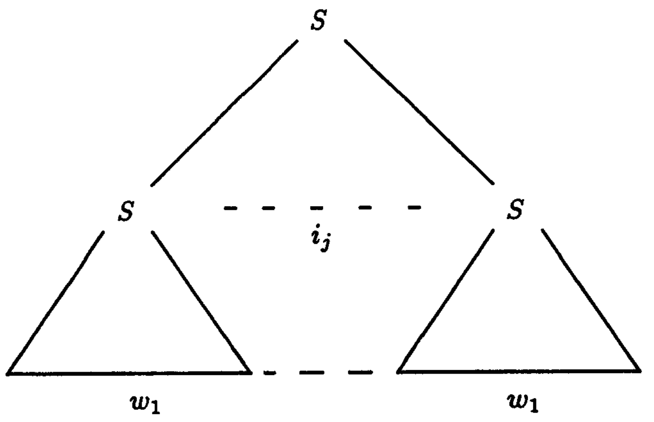


Figure 1: Syntax tree with one nonterminal

Theorem 3.2 *Let $G = (\{S, A\}, \Sigma, P, S)$ and $G' = (\{S', A'\}, \Sigma, P', S')$ be two structurally equivalent invertible E0L grammars. Then, G and G' are isomorphic.*

Proof: Let us first consider productions with a terminal right hand side. Clearly $S \rightarrow w$, $w \in \Sigma^+$, is a production in P if and only if $S' \rightarrow w$ is a production in P'. For, otherwise G and G' would not be structurally equivalent. If $S \rightarrow w$ is a production in P, we cannot have a production $A' \rightarrow w$ in P', because invertibility implies that w can be generated in one step from S'. Similarly, if $S' \rightarrow w$ is in P' we cannot have a production $A \rightarrow w$ in P. Thus, the productions in P and P' with terminal right hand sides and the start symbol as left hand side are isomorphic.

Assume that $A \rightarrow w$ is in P, $w \in \Sigma^+$. Then, $S \rightarrow w$ is not in P. Consider a syntax tree in which this production is applied. Because a syntax tree of the same structure can be generated in G', $A' \rightarrow w$ must be a production in P'; $S' \rightarrow w$ cannot be a production in P', because in this case, w would be generated by a height one syntax tree in G', but not in G. This argument shows that the productions with terminal right hand sides in G and G' are isomorphic.

Let us now consider productions with nonterminal right hand sides. Assume $S \rightarrow X_1 \ldots X_k$, $X_j \in \{S, A\}$, $1 \leq j \leq k$, is a production in P. Then, there exist $w_1, \ldots, w_k \in \Sigma^+$ such that $X_j \Rightarrow^d w_j$, for $1 \leq j \leq k$ and some $d \geq 1$. Now reconstruct the labels of the syntax tree with frontier $w_1 \ldots w_k$ in G'. We obtain $X'_j \Rightarrow^d w_j$, $1 \leq j \leq k$, and $X'_j = S'$ if and only if $X_j = S$, and $X'_j = A'$ if and only if $X_j = A$, for $1 \leq j \leq k$. For, otherwise a sentential syntax tree in G of height d would not have a structurally equivalent counterpart in G' and vice versa. Applying this argument once again we conclude that $S' \to X_1' \dots X_k'$ is in P'. By symmetry, this shows that P and P' must have isomorphic productions with the start symbol as left hand side and nonterminals as right hand sides.

By similar arguments we see that P and P' must also have isomorphic productions with the other nonterminal as left hand side and nonterminals as right hand side. A production $A \to X_1 \dots X_k$ of this form must occur in a terminal derivation in G. Retracing this derivation in G' shows that the corresponding, isomorphic production must occur in G' and vice versa. □

Theorems 3.1 and 3.2 cannot be extended. If we allow more nonterminals we can have invertible, structurally equivalent E0L grammars which are not isomorphic. First, we give an example where both E0L grammars have a different number of nonterminals.

Example 3.1 Recall the E0L grammar G_5 of Example 2.5 which is given by the productions

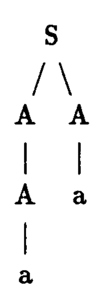
$$S \to AA, \ A \to A|a.$$

A structurally equivalent but nonisomorphic grammar G_6 with three nonterminals is given by the productions

$$S \to AA|BB, A \to B|a, B \to A.$$

G_6 is also an invertible E0L grammar which generates just one word aa by syntax trees of even and odd heights as shown by Figure 2.

Note that G_5 and G_6 are not structurally equivalent if considered as context-free grammars. For, when parallel rewriting is replaced by (asynchronous) sequential rewriting, we can generate syntax trees of different structures with both grammars. The terminal leaves of syntax trees generated by G_6 always have an even depth difference whereas in G_5 we can generate a syntax tree where the terminal leaves have a depth difference of one. For example, the following syntax tree in G_5



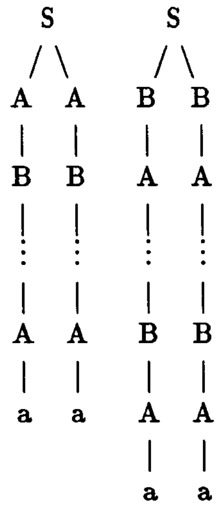


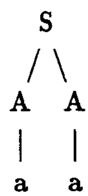
Figure 2: Syntax trees of even and odd heights in G_6

has no structural equivalent counterpart in G_6, if both grammars are considered to be context-free grammars.

Example 3.2 The E0L grammar G_7 with three nonterminals defined by

$$S \rightarrow AA|BB, \ A \rightarrow a, \ B \rightarrow A|B$$

is structurally equivalent to G_6, but not isomorphic to G_6. It is easy to obtain an overview of the sentential syntax trees generated by G_7. The sentential syntax tree



is the only sentential syntax tree of height two in G_7. A sentential syntax tree of height $h > 2$ is of the form shown by Figure 3.

It is easy to see that there is no isomorphism between the productions of G_6 and G_7. Thus, there exist structurally equivalent, nonisomorphic E0L grammars with the same number of nonterminals. In G_6 and G_7 the roles played by the nonterminals A and B are different. We cannot have an A whenever we have a B and vice versa.

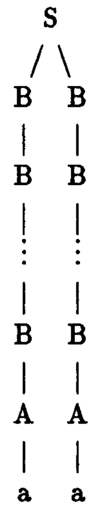


Figure 3: Syntax tree of height greater than two in G_7

Example 3.3 Let the invertible E0L grammar G_8 be given by the productions

$$S \to AA|BB|CC,\ B \to A,\ C \to B|C,\ A \to a.$$

If we map both labels B and C to B, each syntax tree in G_8 becomes a valid syntax tree in G_7. In this case, G_7 can be considered as an homomorphic, but not as an isomorphic, image of G_8. It is obvious how to extend the sequence G_5, G_7, G_8 to obtain an infinite sequence of invertible, structurally equivalent, pairwise nonisomorphic E0L grammars, namely, H_k for $k \geq 1$ is given by

$$\begin{aligned} S &\to X_1X_1|X_2X_2|\ldots|X_kX_k, \\ X_k &\to X_{k-1}|X_k, \\ X_{k-1} &\to X_{k-2},\ \ldots,\ X_2 \to X_1, \\ X_1 &\to a. \end{aligned}$$

Example 3.4 Consider the E0L grammars G_9 and G_{10} defined as follows:

$$G_9 \ :\ S \to AA,\ A \to A|a|b.$$

$$G_{10} \ :\ S \to AA|AB|BA|BB$$

$$A \to B|a$$
$$B \to A|b.$$

$L(G_9) = L(G_{10}) = \{aa, ab, ba, bb\}$; moreover, G_9 and G_{10} are structurally equivalent. Intuitively, in G_{10} A and B play essentially the same roles. It is possible to identify these nonterminals, yielding the simpler grammar G_9 in this case.

In the next section we define context equivalence and show how to effectively reduce an E0L grammar according to this equivalence relation; G_{10} is not context reduced.

4 Context equivalence

The structure of syntax trees of grammars can be encoded via parentheses. Introducing parenthesized versions of E0L grammars allows us to reduce structural equivalence to equivalence.

Given an E0L grammar $G = (N, \Sigma, P, S)$. The *parenthesized version* $G_{()}$ of G is the E0L grammar $G_{()} = (N \cup \{L, R\}, \Sigma \cup \{(,)\}, P_{()}, S)$, where L and R are two new nonterminals and "(" and ")" are new terminal symbols. Matching nonterminal L-R pairs represent matching pairs of parentheses. For $\alpha \in N^+$, $X \to L\alpha R$ is a production in $P_{()}$ if and only if $X \to \alpha \in P$ and, for $w \in \Sigma^+$, $X \to (w)$ is in $P_{()}$ if and only if $X \to w \in P$. Furthermore, $P_{()}$ contains the additional productions $L \to L|($ and $R \to R|)$. Rewriting and other related notions from Section 2 are extended to parenthesized versions of grammars in the obvious way. Of course, if $L(G) \subseteq \Sigma^+$, then $L(G_{()}) \subseteq (\Sigma \cup \{(,)\})^+$. Observe that for a word $x \in L(G_{()})$ all terminal symbols from Σ in x are surrounded by the same number of matching pairs of parentheses. This number is equal to the length of the derivation of x in $G_{()}$ from the start symbol. Obviously, two E0L grammars G and G' are structurally equivalent if and only if their corresponding parenthesized versions generate the same language $L \subseteq (\Sigma \cup \{(,)\})^+$.

A parenthesized nonterminal *context* of an E0L Grammar $G = (N, \Sigma, P, S)$ is a sentential form of the parenthesized version $G_{()}$ in which one occurrence of a nonterminal symbol from N is replaced by an underscore. We call such a word $\alpha \in (N \cup \{L, R\} \cup \{\_\})^+$ simply a *context*. Given a context α and a nonterminal A, $\alpha[A]$ denotes the word obtained by replacing the underscore in α with A.

Let $S \Rightarrow^d \alpha[A]$ in $G_{()}$. $\alpha[A]$ can be identified with a sentential syntax tree of height d in G from which all labels except for the labels at its frontier have been removed. The sequence of labels at the frontier of this tree is denoted by $frontier(\alpha[A])$. Clearly, $S \Rightarrow^d frontier(\alpha[A])$ in G if and only if $S \Rightarrow^d \alpha[A]$ in $G_{()}$. Given an E0L grammar $G = (N, \Sigma, P, S)$ and two nonterminals A

and B, we say that A and B are *d-context equivalent*, denoted by $A \equiv^d B$, if for all contexts α, we have in the parenthesized version $G_{()}$ of G

$$S \Rightarrow^d \alpha[A] \text{ if and only if } S \Rightarrow^d \alpha[B].$$

This definition in particular implies that $A \equiv^0 B$ is equivalent to ($A = S$ if and only if $B = S$). A and B are said to be *context equivalent*, denoted by $A \equiv B$, if $A \equiv^d B$, for all integers $d \geq 0$. Observe that $A \equiv S$ for some nonterminal A implies $A = S$. This reflects the fact that at the root of each syntax tree in G we can have only one nonterminal, the sentence symbol S, but no other nonterminal. We say an E0L grammar is *context reduced* if whenever two nonterminals are context equivalent they are equal.

Example 4.1 Grammar G_{10} of Example 3.4 is not context reduced. It is easy to prove by induction on d that A and B are d-context equivalent, for all integers $d \geq 0$.

Given an E0L grammar $G = (N, \Sigma, P, S)$, context equivalence of nonterminals can also be characterized in a different manner (cf. [1] for the case of context-free grammars). For each context α and integer $d \geq 0$, we define

$$N_d^\alpha = \{X : S \Rightarrow^d \alpha[X] \text{ in } G_{()}\}$$

to be the set of nonterminals *d-distinguishable in context* α. The definition implies that $N_0^\alpha = \{S\}$, for all α, that is, $\{S\}$ is 0-distinguishable, for all α. A set M of nonterminals is said to be *d-distinguishable* if there is a context α such that $M = N_d^\alpha$. M is *strictly d-distinguishable* if M is d-distinguishable but not d'-distinguishable for any $d' < d$. M is *distinguishable* if there is a $d \geq 0$ such that M is d-distinguishable.

Theorem 4.1 *Let $G = (N, \Sigma, P, S)$ be an E0L grammar and $A, B \in N$. Then, $A \equiv B$ if and only if for all distinguishable sets $M \subseteq N$, A is in M if and only if B is in M.*

Proof: *if:* If $V = \{S\}$, then ($A \in V$ if and only if $B \in V$) immediately implies ($A = S$ if and only if $B = S$). Therefore, we have $A \equiv^0 B$. Now consider an arbitrary context α and an integer $d \geq 1$. We have to show

$$S \Rightarrow^d \alpha[A] \text{ if and only if } S \Rightarrow^d \alpha[B].$$

Consider the set N_α^d of nonterminals d-distinguishable in context α, namely,

$$\{X : S \Rightarrow^d \alpha[X]\}.$$

By definition N_α^d is d-distinguishable, therefore distinguishable. By assumption this set contains A if and only if it contains B. This immediately gives the hypothesis.

only if: Let $A \equiv B$. This implies by definition ($A = S$ if and only if $B = S$) and, for all contexts α, and for all $d \geq 1$, ($S \Rightarrow^d \alpha[A]$ if and only if $S \Rightarrow^d \alpha[B]$). If $M = \{S\}$, then we immediately have $A \in M$ if and only if $B \in M$. Hence, let M be a distinguishable subset of N and $M \neq \{S\}$. Then, there exists a context α and an integer $d \geq 1$ such that $M = N_d^\alpha$. In particular $A \equiv B$ implies ($S \Rightarrow^d \alpha[A]$ if and only if $S \Rightarrow^d \alpha[B]$); hence, ($A \in M$ if and only if $B \in M$). □

It is clear that there exist only finitely many different distinguishable subsets of nonterminals for a given grammar. If the distinguishable sets can be effectively determined the equivalence relation $\equiv$ can also be computed effectively. The question is whether or not we have to consider arbitrary long derivations in order to find all distinguishable sets of nonterminals. The following lemmas show that this is not necessary.

Lemma 4.2 *Let $G = (N, \Sigma, P, S)$ be an invertible n-complete E0L grammar and $M \subseteq N$ strictly d-distinguishable. Then, there exists an $M' \neq M$, $M' \subseteq N$, which is strictly $(d-1)$-distinguishable.*

Proof: Let $M = N_d^\alpha = \{X : S \Rightarrow^d \alpha[X]\}$, for some $d > 1$ and for some context α such that, for each $d' < d$ and each context β, $M \neq N_{d'}^\beta$. Consider a syntax tree for $frontier(\alpha[X])$ in G of height $d > 1$, for some $X \in M$; see Figure 4.

All nonterminals in M which may occur instead of X at the same position on level d in $frontier(\alpha[X])$ are generated by nonterminals which occur at the same position on level $d-1$. Because G is invertible, all elements of $frontier(\alpha[X])$ on level d have a uniquely determined predecessor on level $d-1$. Thus, the context α uniquely determines a context β. Replace each innermost $L-R$ pair surrounding a right hand side of a production in P by its left hand side and, moreover, replace the innermost $L-R$ pair containing the underscore in α by an underscore in β. Let us assume that the innermost $L-R$ pair in α containing the underscore is of the form $L\alpha_l\_\alpha_r R$. Then, it is clear that

$$M = N_d^\alpha = \{X : Y \rightarrow L\alpha_l X \alpha_r R \text{ is a production in } P_{()} \text{ and } Y \in N_{d-1}^\beta\}.$$

We claim that N_{d-1}^β is strictly $(d-1)$-distinguishable. Assume that this is not true. This implies that there is a context β' and a $d' < d-1$ such that

$$N_{d-1}^\beta = N_{d'}^{\beta'} = \{Y : S \Rightarrow^{d'} \beta'[Y] \text{ in } G_{()}\}.$$

Because G is n-complete, we can add one parallel derivation step to the derivation of length d' in $G_{()}$ of the context β'. Each nonterminal in β' is

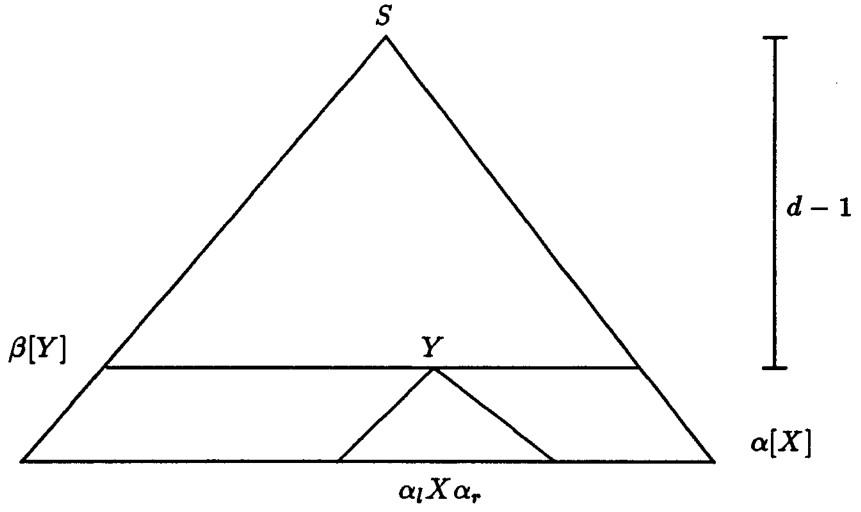


Figure 4: Syntax tree of height $d > 0$ for $frontier(\alpha[X])$

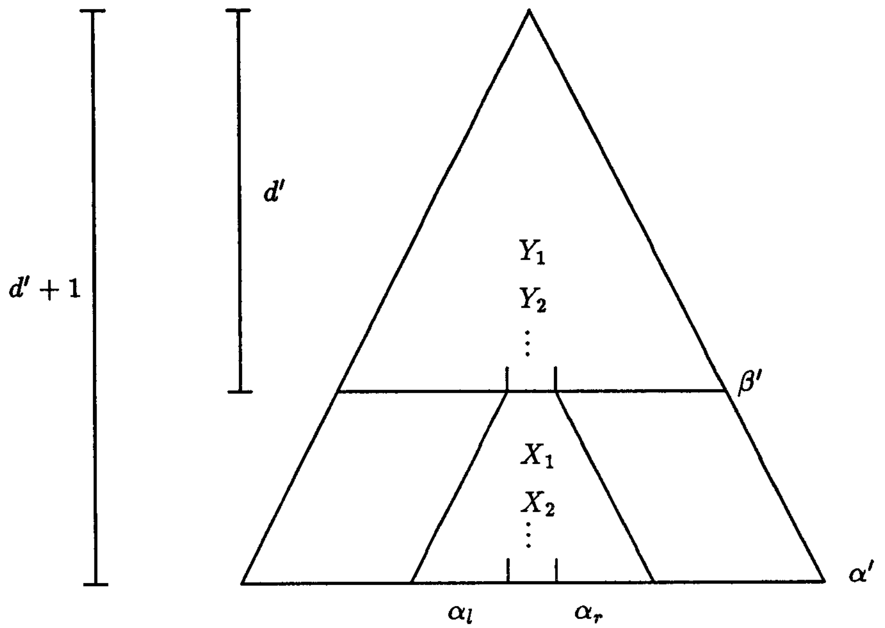


Figure 5: Definition of context α'

replaced by the right hand side of a corresponding production in $P_{()}$ and the underscore is replaced by $L\alpha_l\_\alpha_r R$. This yields a context α'; see Figure 5. We obtain $M = N_d^\alpha = N_{d'+1}^{\alpha'}$, where $d'+1 < d$; this contradicts the assumption that M is strictly d-distinguishable. Hence we have proved our claim that N_{d-1}^β is strictly $(d-1)$-distinguishable. Therefore, the lemma holds with $M' = N_{d-1}^\beta$. □

As an immediate consequence of Lemma 4.2 we infer that in $G_{()}$ only derivations of length $d = 2^n$, $n = |N|$, have to be considered in order to determine all distinguishable sets of nonterminals. The argument used in the proof of Lemma 4.2, however, requires that each derivation of a nonterminal word in the n-complete E0L grammar G can be extended by one parallel derivation step. The following lemma shows that it is not necessary to assume that an E0L grammar is n-complete in order to infer that only finitely many derivations have to be considered in order to determine the distinguishable sets of nonterminals.

Lemma 4.3 *Let $G = (N, \Sigma, P, S)$ be an invertible E0L grammar with $n = |N|$ and let $M \subseteq N$, $|M| = m$, be a strictly d-distinguishable set of nonterminals. Then, $d < (n^m + 1)(2^{n^m} + 1)$.*

Proof: Assume the contrary, that is, $M = N_d^\alpha$, for some context α and some minimal $d \geq (n^m + 1)(2^{n^m} + 1)$, and

$$M = N_d^\alpha = \{X : S \Rightarrow^d \alpha[X] \text{ in } G_{()}\} = \{A_1, \ldots, A_m\}.$$

Because G is invertible we can uniquely associate a labeled tree of height d and structure α to each $A_j \in M$, $1 \leq j \leq m$. Reconstruct the labels at all internal nodes in $\alpha[A_j]$ bottom up to obtain a sentential syntax tree for $frontier(\alpha[A_j])$ in G, $1 \leq j \leq m$. All associated syntax trees in G for $frontier(\alpha[X])$, $X \in M$, of height d have the same structure. They all have the label S at the root and the same labels at the frontier with only one exception, the position of the underscore in α. There $A_1, \ldots, A_m$ appear. However, the labels of internal nodes may be different. We call the root-to-leaf path connecting the root and the position of the underscore in α the *designated* path. With each internal node in α we can associate an m-tuple $(X_1, \ldots, X_m)$ of nonterminals as follows. $(X_1, \ldots, X_m)$ is associated to a node p if and only if X_j occurs in the syntax tree for $frontier(\alpha[A_j])$ of height d and structure α at node p, $1 \leq j \leq m$. Because there are only n^m different m-tuples of nonterminals and $d \geq (n^m + 1)(2^{n^m} + 1)$ there must be at least $2^{n^m} + 1$ different levels in α such that the same m-tuple is associated to the nodes on the designated path on these levels. Because there are at most 2^{n^m} different subsets of m-tuples of nonterminals there must be at least two levels in α such that not only the same m-tuple of nonterminals is associated to the nodes on the designated path on these levels but also the sets of m-tuples of nonterminals associated to the other nodes on these two levels are identical. This implies that there are integers d_1, d_2, and d_3 such that $d_1 + d_2 + d_3 = d$ and (i), (ii), and (iii) hold.

(i) For each j, $1 \leq j \leq m$, the labels of the nodes on the designated path on levels d_1 and $d_1 + d_2$ below the root in the syntax tree of $frontier(\alpha[A_j])$ of height d and structure α are identical.

(ii) For each j, $1 \leq j \leq m$, there are nonterminal words w_j and w'_j such that

$$S \Rightarrow^{d_1} w_j \Rightarrow^{d_2} w'_j \Rightarrow^{d_3} frontier(\alpha[A_j]),$$

where w_j and w'_j are words over the same set N_j of nonterminals. N_j is obtained by taking the set of j-th components in the set of all m-tuples of nonterminals on levels d_1 or $d_1 + d_2$ in α.

(iii) Let p and q be nodes on levels d_1 and $d_1 + d_2$ in α, respectively, to which the same m-tuple of nonterminals is associated. Then for all j, $1 \leq j \leq m$, the syntax tree of $frontier(\alpha[A_j])$ of height d and structure α has identical nonterminals at the nodes p and q.

Therefore, we can cut out the part in between levels d_1 and $d_1 + d_2$ in α as follows. Replace each subtree of height $d_2 + d_3$ with root on level d_1 by the subtree of height d_3 with root on level $d_1 + d_2$ with the same m-tuple of nonterminals associated to its root. Because the sets of m-tuples of nonterminals associated to the nodes on these two levels are identical we will find for each node on level d_1 an appropriate node on level $d_1 + d_2$. In particular, the subtree with root node on the designated path on level d_1 is replaced by the subtree with root node on the designated path on level $d_1 + d_2$. This situation is illustrated by Figure 6. In this way we obtain a context α' such that $N_d^\alpha = N_{d_1+d_2}^{\alpha'} = M$. Because $d_1 + d_2 < d$ we have a contradiction. □

Let $G = (N, \Sigma, P, S)$ be an invertible E0L grammar. We will now show how to reduce G according to the equivalence relation "$\equiv$".

Theorem 4.4 *Let $G = (N, \Sigma, P, S)$ be an invertible E0L grammar. We can effectively construct a context reduced E0L grammar $\bar{G} = (\bar{N}, \Sigma, \bar{P}, \bar{S})$ which is structurally equivalent to G. The construction maintains invertibility, if the given E0L grammar G is n-complete.*

Proof: For each $X \in N$ the equivalence class of X with respect to $\equiv$ will be denoted by $\bar{X}$. Define $\bar{G} = (\bar{N}, \Sigma, \bar{P}, \bar{S})$, where

$$\bar{N} = \{\bar{X} : X \in N\}, \ \bar{P} = \{\bar{A} \rightarrow \bar{\alpha} : A \rightarrow \alpha \in P\},$$

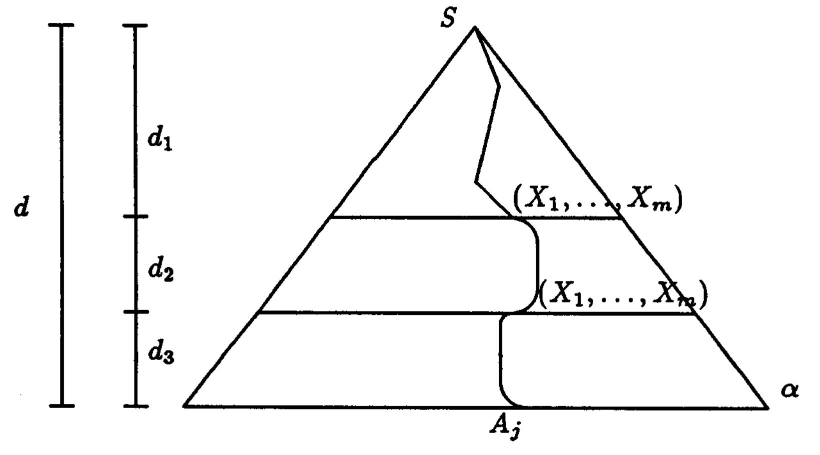


Figure 6: Levels with identical sets of m-tuples

and $\bar{\alpha}$ denotes the word over the alphabet $\bar{N} \cup \Sigma$ which is obtained by replacing each nonterminal by its equivalence class with respect to $\equiv$; we say $\bar{\alpha}$ *corresponds* to α. Because of Lemmas 4.2 and 4.3, $\bar{G}$ can be effectively constructed from G. Note that $\bar{\alpha} = \bar{\beta}$ and $\alpha \neq \beta$ is possible, if α and β contain different but context equivalent nonterminals. In other words, the same word in $\bar{G}$ may correspond to different words in G. However, $\alpha = \beta$ if $\bar{\alpha} = \bar{\beta}$ and both α and β are terminal words.

It is clear that each derivation $S \Rightarrow^* \alpha$ in G can be transformed into a derivation of the same length and with a syntax tree of the same structure in $\bar{G}$. Each derivation step in G becomes a derivation step in $\bar{G}$ by taking corresponding words. This yields a derivation $\bar{S} \Rightarrow^* \bar{\alpha}$ in $\bar{G}$. We will now show that conversely for each derivation in $\bar{G}$ there is a derivation of the same length and structure in G. More precisely, we will show by induction on d, $d \geq 0$, that the following claim holds.

Claim *Let $\bar{S} \Rightarrow^d \bar{\alpha}$ in $\bar{G}$, $\bar{\alpha} \in (\bar{N}^+ \cup \Sigma^+)$, and let γ be an arbitrary word such that $\bar{\gamma} = \bar{\alpha}$. Then, $S \Rightarrow^d \gamma$ in G and both syntax trees are structurally identical.*

Proof of Claim: The proof is by induction on the length of derivations. We consider first a derivation of length 0. Let $\bar{S} \Rightarrow^0 \bar{\alpha}$ in $\bar{G}$ and γ such that $\bar{\gamma} = \bar{\alpha}$. Then, $\bar{\alpha} = \bar{S}$ and $\gamma = A$, where $A \equiv S$. By the definition of $\equiv$ this implies $A = S$. Hence, we have $S \Rightarrow^0 \gamma$ in G.

Consider a derivation $\bar{S} \Rightarrow^d \bar{\beta} \Rightarrow \bar{\alpha}$ of length $d+1$ in $\bar{G}$. We first show that there is a γ that satisfies the claim and, then, in the next paragraph we show that the claim holds for all γ such that $\bar{\gamma} = \bar{\alpha}$. Let $\bar{\beta} = \bar{B}_1 \ldots \bar{B}_k$, for some k and $B_1, \ldots, B_k \in N$. Then, $\bar{\alpha}$ is obtained by parallel rewriting all $\bar{B}_j$ according to productions $\bar{B}_j \rightarrow \bar{a}_j$ in $\bar{P}$; that is $\bar{\alpha} = \bar{\alpha}_1 \ldots \bar{\alpha}_k$, for some $\alpha_1, \ldots, \alpha_k$. By the definition of P, $\bar{B}_j \rightarrow \bar{\alpha}_j$ are productions in $\bar{P}$ only if $B_j \rightarrow \alpha_j$ are productions in P. $\bar{\beta}$ corresponds to $B_1 \ldots B_k$; hence, by the inductive assumption, $B_1 \ldots B_k$ can be derived in G by a derivation of length d with a syntax tree of the same structure as the syntax tree of $\bar{S} \Rightarrow^d \bar{\beta}$ in $\bar{G}$. Rewriting $B_1, \ldots, B_k$ in parallel by $\alpha_1, \ldots, \alpha_k$ yields a word γ such that $\bar{\gamma} = \bar{\alpha}$ and γ can be derived in $(d+1)$ steps in G with a syntax tree which is of the same structure as the syntax tree for $\bar{\alpha}$ in $\bar{G}$. Thus, we know that $\bar{S} \Rightarrow^{d+1} \bar{\alpha}$ in $\bar{G}$ implies that there *exists* a word γ such that $\bar{\gamma} = \bar{\alpha}$ and $S \Rightarrow^{d+1} \gamma$ in G and the two syntax trees have the same structure.

Now assume that γ' is a word which also corresponds to $\bar{\alpha}$; that is, $\bar{\gamma}' = \bar{\alpha}$. We have to show that $S \Rightarrow^{d+1} \gamma'$ in G and the syntax tree is of the same structure as the one in the derivation of $S \Rightarrow^{d+1} \gamma$. We may assume that both γ and γ' are nonterminal words. (For, otherwise, they must be identical). Let $\gamma = X_1 \ldots X_l$; then $\bar{\gamma} = \bar{\gamma}'$ implies $\gamma' = X_1' \ldots X_l'$, where $X_j \equiv X_j'$, for $1 \leq j \leq l$. By the definition of the relation $\equiv$ we infer that with γ also the following sequence of words is derivable in $(d+1)$ steps in G by syntax trees of identical structure:

$$\begin{aligned} \gamma \; = \; & X_1 X_2 \ldots X_l, \\ & X_1' X_2 \ldots X_l, \\ & \vdots \\ & X_1' \ldots X_j' X_{j+1} \ldots X_l \\ & \vdots \\ & X_1' \ldots X_l' = \gamma' \end{aligned}$$

This completes the induction and the proof of the claim. □

The claim in particular implies that the same terminal words are derivable in G and $\bar{G}$ with sentential syntax trees of identical structures. Therefore, G and $\bar{G}$ are structurally equivalent.

Next, we show that $\bar{G}$ is context reduced. Assume that $M, M' \subseteq N$ are two nonterminals in $\bar{G}$ and $M \equiv M'$. Choose $X \in M$ and $X' \in M'$. Let α be an arbitrary context for G. $\bar{\alpha}$ corresponds to α and is a context for $\bar{G}$. Moreover $\bar{\alpha}[M]$ corresponds to $\alpha[X]$ and $\bar{\alpha}[M']$ corresponds to $\alpha[X']$. As we have shown above, corresponding words are derivable in G and $\bar{G}$ by derivations of the same length and with syntax trees of the same structure. Therefore, we have, for all d,

$$S \Rightarrow^d \alpha[X] \text{ in } G_{()} \text{ if and only if } \bar{S} \Rightarrow^d \bar{\alpha}[M] \text{ in } \bar{G}_{()},$$

and

$$S \Rightarrow^d \alpha[X'] \text{ in } G_{()} \text{ if and only if } \bar{S} \Rightarrow^d \bar{\alpha}[M'] \text{ in } \bar{G}_{()}.$$

Hence, we can infer from $M \equiv M'$ that $X \equiv X'$. This, in turn, implies that $\bar{X} = M = M' = \bar{X}'$. Thus, we see that context equivalent nonterminals in $\bar{G}$ must be identical; that is, $\bar{G}$ is context reduced.

Finally, we show that the construction maintains invertibility, if the given E0L grammar is n-complete; that is we show that $\bar{G}$ is invertible if G is invertible and n-complete. Let $\bar{X} \rightarrow \bar{w}_1$ and $\bar{Y} \rightarrow \bar{w}_2$ be two productions in $\bar{G}$ with equal right hand sides; that is $\bar{w}_1 = \bar{w}_2$, and let these productions correspond to two productions $X \rightarrow w_1$ and $Y \rightarrow w_2$ in G. If w_1 and w_2 are both terminal words, they must be identical and, moreover, $X = Y$ because G is invertible. Hence, we may assume that w_1 and w_2 are nonterminal words. Consider an arbitrary context α such that $S \Rightarrow^d \alpha[X]$ in G. Because G is n-complete we can extend this derivation by one parallel derivation step. We replace X by w_1 and rewrite all other nonterminals by nonterminal words arbitrarily. We obtain some $\beta[w_1]$ such that $S \Rightarrow^d \alpha[X] \Rightarrow \beta[w_1]$ in $G_{()}$. Consider $\gamma_1 = \mathit{frontier}(\beta[w_1])$. Then, we know that $S \Rightarrow^{d+1} \gamma_1$ and $\beta[w_1]$ describes the structure of this syntax tree in G. As we have just shown, γ_1 can be derived in G if and only if $\bar{\gamma}_1$ can be derived in $\bar{G}$ with syntax trees of identical structure if they exist. Replacing w_1 by w_2 in $\beta[w_1]$ gives a word $\beta[w_2]$ in $G_{()}$ such that, for $\gamma_2 = \mathit{frontier}(\beta[w_2])$, we have $\bar{\gamma}_2 = \bar{\gamma}_1$, because $\bar{w}_1 = \bar{w}_2$ and all other letters in γ_1 and γ_2 are identical. Hence, γ_1 can be derived in G if and only if γ_2 can be derived in G and both syntax trees have the same structure if they exist. Because G is invertible, we can infer that in $G_{()}$ $S \Rightarrow^d \alpha[Y]$ holds. Thus, we see that for each context α and for each d, $S \Rightarrow^d \alpha[X]$ if and only if $S \Rightarrow^d \alpha[Y]$ in $G_{()}$. This implies $X \equiv Y$ and, therefore, $\bar{X} = \bar{Y}$. This shows that there cannot be two different productions in $\bar{G}$ with equal right hand sides. □

Remark: The proof of Theorem 4.4 can easily be carried over to the case of E0L grammars with more than one sentence symbol. It is in this form that the theorem is used in [2].

Eliminating context equivalent nonterminals by reducing a grammar according to the relation $\equiv$ does *not* uniquely determine an E0L grammar up to isomorphism. There exist structurally equivalent invertible context reduced E0L grammars which are not isomorphic. Consider the grammar G_6 of Examples 3.1 given by the productions

$$S \rightarrow AA|BB, A \rightarrow B|a, B \rightarrow A.$$

Clearly, $A \not\equiv S$ and $B \not\equiv S$. Consider the context $\alpha = L\_AR$; then, we have $S \Rightarrow^1 \alpha[A]$ in $G_{6,()}$ but $S \Rightarrow^1 \alpha[B]$ does *not* hold; hence, $A \not\equiv B$. This

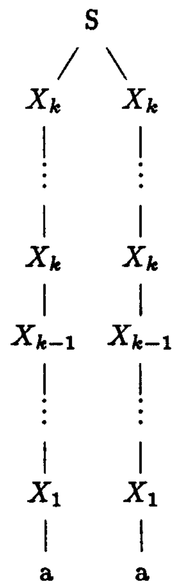


Figure 7: Syntax tree of height $dk + r + 1$ in $G(k)$

shows that G_6 is context reduced. The structurally equivalent grammar G_5 of Example 2.5 given by the productions

$$S \rightarrow AA,\ A \rightarrow A|a$$

is also context reduced but not isomorphic to G_6. We can even have structurally equivalent invertible, context reduced grammars with the same number of nonterminals. Consider the E0L grammar G_7 of Example 3.2. It is easy to see that G_7 is context reduced. As we already know, G_7 is invertible and structurally equivalent to G_6. This example can be generalized to provide an infinite sequence of grammars. Consider the E0L grammar $G(k)$ with $k + 1$ nonterminals given by the productions

$$S \rightarrow X_1X_1|X_2X_2|\ldots|X_kX_k,$$

$$X_k \rightarrow X_{k-1},\ldots,X_2 \rightarrow X_1,$$

$$(*)X_k \rightarrow X_k,$$

$$X_1 \rightarrow a.$$

Now, $G(k)$ is structurally equivalent to the grammar $G'(k)$, which has the same productions except for the production marked by (*). Instead of this

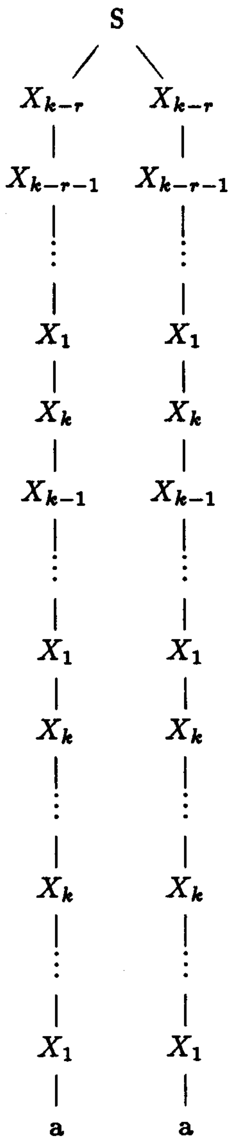


Figure 8: Syntax tree of height $dk + r + 1$ in $G'(k)$

production $G'(k)$ contains $X_1 \to X_k$. It is easy to check that both grammars are invertible, context reduced, structurally equivalent but not isomorphic.

The syntax trees of height $dk + r + 1$ in $G(k)$, $d \geq 1$, $0 \leq r < k$, are displayed in Figure 7. Figure 8 shows the syntax trees of the same height $dk + r + 1$ in $G'(k)$.

5 Conclusions

As we have demonstrated in the preceding sections, structural equivalence of E0L grammars is decidable in some special cases. However, we have been unable to obtain a general decidability result in contradistinction to the case of context-free grammars. We have imposed additional properties on E0L grammars which preserve structural equivalence in order to achieve a new normal form. In the case of context-free grammars reducing an invertible grammar according to context equivalence has the effect that the resulting grammar is uniquely determined up to isomorphism. We have seen that this does not hold for E0L grammars. The reason is, of course, that the superficially similar notions of context equivalence for context-free and E0L grammars are, in fact, completely different. This difference results from the different rewriting mechanisms.

Thus, the basic open question resulting from our investigations is: Is there an effectively decidable equivalence relation between the nonterminals of an E0L grammar such that reducing the grammar according to this relation preserves structural equivalence and yields a grammar which is uniquely determined up to isomorphism?

References

[1] R. McNaughton. Parenthesis grammars. *Journal of the ACM*, 14:490–500, 1967.

[2] Th. Ottmann and D. Wood. Defining families of trees with E0L grammars. University of Waterloo, 1989.

[3] G. Rozenberg and A. Salomaa. *The Mathematical Theory of L Systems.* Academic Press, New York, 1980.

[4] A. Salomaa. *Formal Languages.* Academic Press, New York, 1973.

[5] D. Wood. *Theory of Computation.* John Wiley & Sons, New York, 1987.