Publ. RIMS, Kyoto Univ.
9 (1973), 61-92

On Derivation Trees of Indexed Grammars
—An Extension of the uvwxy-Theorem—

By

Takeshi HAYASHI

Abstract

In this paper, the uvwxy-theorem of context-free languages is extended to the case of indexed languages. Applying the extended theorem, it is shown that the finiteness problem for the indexed languages is solvable and certain languages such as $\{a^{n!} \mid n \geqq 1\}$ and $\{(\$ w)^{|w|} \mid w \in \{a, b\}^*\}$ are not indexed languages.

Introduction

An extension of the well-known uvwxy-theorem (Bar-hillel et al. [2]) to the one-way stack languages has been given by Ogden [4]. This paper considers an extension to Aho's [1] indexed languages, the family of which properly includes all one-way stack languages.

Our extension is considered to be a kind of intercalation theorem, following Ogden's terminology. He has called his extended uvwxy-theorem an intercalation theorem, since it asserts that, given a string x in a one-way stack language L, it is possible under certain hypotheses to intercalate other strings into x and still stay in the language L.

This paper, however, treats the derivation trees instead of strings. Namely we focus our attention on the set $\mathscr{T}(G)$ of all terminal derivation trees of an indexed grammar G. Now our main theorem states that if a given terminal derivation tree γ in $\mathscr{T}(G)$ is big enough, then we can generate new terminal derivation trees in $\mathscr{T}(G)$ by the insertion of other trees into γ.

In Section 1, to treat the derivation trees precisely, we adopt and

Received September 22, 1972.

extend the notations developed by Brainerd [3] and Takahashi [6]. Also we review Aho's definition of an indexed grammar and give related definitions. Sections 2~4 are devoted to the detailed discussions concerning the terminal derivation trees of an indexed grammar. In Section 2, we determine a constant k depending on a given indexed grammar G with the following property. If the number of maximal nodes of a terminal derivation tree γ in $\mathscr{T}(G)$ is greater than k, then there is a decomposition of γ which assures the existence of parts of γ that can be intercalated. Using the result in Section 2, we prove a lemma in Section 3 which actually describes how to insert other trees into γ and get new terminal derivation trees. In Section 4, making use of the lemma in the previous section,we give a lemma which asserts the strict growth of the maximal nodes of the new terminal derivation trees obtained from γ. Our main theorem is given in Section 5. Also the applications of the theorem are investigated. First it is shown that the finiteness problem for the indexed languages is solvable. Next we give a theorem which states that certain languages such as $\{a^{n!} \mid n \geq 1\}$ are not indexed languages. Finally it is shown that the language $L_\Sigma = \{(\$w)^{|w|} \mid w \in \Sigma^*\}$ is not an indexed language, where Σ is an arbitrary alphabet not including \$.

1. Preliminaries

In this section we give and review basic definitions concerning trees and indexed grammars. First we define the trees over an alphabet with certain operations on them. Newt we review the definition of an indexed grammar given by Aho [1], and reformulate the related concepts in terms of trees.

A. Trees

Definition 1.1. The *universal tree domain* J^* is the free monoid generated by J, where J is the set of all positive integers. We denote the concatenation operator by $\cdot$ and the identity by 0. A finite subset D of J^* is said to be a *tree domain* if D satisfies the following conditions:

(a) If $p \cdot q$ is in D, then p is in D.

(b) If $p \cdot j$ is in D and $1 \leqq i \leqq j$, then $p \cdot i$ is in D.

We call an element of D a *node*. The condition (a) implies that if D is not empty, 0 is always is in D. The node 0 is called the *root*. Now we introduce two relations $<$ and $\lessdot$ on J^*, representing the ancester-descendant relation and the left-to-right relation respectively.

Definition 1.2. For p, q in J^*,

(a) $p \leqq q$ means that there exists r in J^* such that $q = p \cdot r$;

(b) $p < q$ means that $p \leqq q$ and $p \neq q$;

(c) $p \lessdot q$ means that there exist r in J^*, i in J and j in J such that $i < j$, $r \cdot i \leqq p$ and $r \cdot j \leqq q$.

For $p, q \in D$ (D is a tree domain), when $p < q$ holds, a node q is called a *descendant* of a node p, and when $p \lessdot q$ holds, q is to the *right* of p.

Remark 1.1. It can be easily verified that for $p, q \in J^*$ (also for D), one and only one of the followings holds:

$$p = q,\ p < q,\ q < p,\ p \lessdot q, \text{ or } q \lessdot p.$$

Definition 1.3. Let Σ be a set of symbols. A function $\gamma\colon D \to \Sigma$ is called a *tree over* Σ, where D is a tree domain. $\gamma(p)$ is called the *label* of a node p. From now on we denote the domain D of γ by D_γ, and identify a map γ with its graph $\{(p, \gamma(p)) \mid p \in D_\gamma\}$. We write $\gamma(p) = a$ when (p, a) is in γ. Δ_Σ denotes the set of all trees over Σ.

Definition 1.4. For a tree $\gamma \in \Delta_\Sigma$, the *front* $\hat{\gamma}$ of γ is a subset of γ such that:

$$\hat{\gamma} = \{(p, a) \in \gamma \mid \text{ for any } i \in J,\ p \cdot i \notin D_\gamma\}.$$

Namely $\hat{\gamma}$ is the set of all nodes of γ which have no descendants in D_γ. Such nodes are called *maximal nodes* (*of* γ). The sequence $\langle (p_0, a_0), (p_1, a_1), \ldots, (p_n, a_n) \rangle$ of elements of γ in Δ_Σ is called a *chain of* γ if

$$p_0 = 0,\ (p_n, a_n) \in \hat{\gamma},\ p_i = p_{i-1} \cdot j_i \text{ for some } j_i \in J\ (i = 1, \ldots, n).$$

B. Operations on Trees

Definition 1.5. For $\gamma \in \Delta_\Sigma$, we set:

(a) $\gamma/p=\{(q, a)|(p\cdot q, a)\in\gamma\}$ for $p\in D_\gamma$, which is called the *subtree of γ at p*;

(b) $p\cdot\gamma=\{(p\cdot q, a)|(q, a)\in\gamma\}$ for $p\in J^*$, which defines the operation to attach γ under the node p;

(c) $\gamma\backslash E=\{(p, a)\in\gamma\,|\,q\nless p$ for any $q\in E\}$ for $E\subseteq D_\gamma$, whih defines the operation to take away the subtrees under $q\in E$; and

(d) for $p\in D_\gamma$ and $E\subseteq D_\gamma$,

$$\gamma<p, E>=\begin{cases}(\gamma\backslash E)/p & \text{if } E\neq\phi \text{ and } p<q \text{ holds for any } q\in E,\\ \gamma/p & \text{if } E=\phi,\\ \text{undefined otherwise,}^{1)} & \end{cases}$$

which defines the subtree of γ at p after taking away the subtrees under $q\in E$.

Note that γ/p and $\gamma\backslash E$ are always in Δ_Σ.

Definition 1.6. For γ and $\tau_0,\dots,\tau_n\in\Delta_\Sigma$ and $p_i\in D_\gamma(i=0,\dots,n)$ satisfying $p_i\nless p_{i+1}(i=0,\dots,n-1)$ and $\tau_i(0)=\gamma(p_i)$ $(i=0,\dots,n)$, we set

$$\gamma\begin{bmatrix}p_0,\dots,p_n\\ \tau_0,\dots,\tau_n\end{bmatrix}=(\gamma\backslash E)\cup\bigcup_{i=0}^{n}p_i\cdot\tau_i,\ \text{where } E=\{p_0,\dots,p_n\},$$

which is the result of inserting $\tau_0,\dots,\tau_n$ into the places of $p_0,\dots,p_n$.

Definition 1.7. The *yield function* g is a function from Δ_Σ into Σ^* defined recursively as follows [2]

(1) $g(\gamma)=\gamma(0)$ if $D_\gamma=\{0\}$,

(2) $g(\gamma)=g(\gamma/1)\dots g(\gamma/j)$ if $1,\dots,j\in D_\gamma$, $j+1\notin D_\gamma$.

Note that the function g makes the string of labels attached to the maximal nodes of γ keeping the left-to-right order.

C. Trees of an Indexed Grammar

Definition 1.8. An *indexed grammar* is a 5-tuple, $G=(N, T, F, P, S)$,

1) If E is a singleton set, say $E=\{q\}$, then we write $\gamma\backslash q$ and $\gamma<p,q>$ instead of $\gamma\backslash\{q\}$ and $\gamma<p, \{q\}>$ respectively.

2) Let Σ be a set. Σ^* is the set of all finite strings of elements of Σ, including ε, the empty string.

in which:

(a) N is a finite nonempty set of symbols called the *nonterminal alphabet*.

(b) T is a finite set of symbols such that $N \cap T = \phi$ and called the *terminal alphabet*.

(c) F is a finite set each element of which is a finite set of ordered pairs of the form (A, χ), where A is in N and χ is in $(N \cup T)^*$. An element f in F is called an *index*. An ordered pair (A, χ) in f is written $A \rightarrow \chi$ and is called an *index production* in f.

(d) P is a finite set of ordered pairs of the form (A, α) with A in N and α in $(NF^* \cup T^*)$. Such a pair is usually written $A \rightarrow \alpha$; it is called a *production*.

(e) S, the *sentence symbol*, is a distinguished symbol in N.

Now we must define a derivation in an indexed grammar G. This is done merely by translating Aho's original definition into our system. Namely, let $G = (N, T, F, P, S)$ be an indexed grammar, and let V be equal to $NF^* \cup T \cup \{\varepsilon\}$, then we consider Δ_V (all the trees over V) and define a relation $\vdash_{\overline{G}}$ on Δ_V whose reflexive and transitive closure $\vdash^{*}_{\overline{G}}$ corresponds to the derivation. [3]

Definition 1.9. Let $G = (N, T, F, P, S)$ be an indexed grammar, [4] a relation $\vdash_{\overline{G}}$ on $\Delta_V(V = NF^* \cup T \cup \{\varepsilon\})$ is defined as follows. For $\gamma, \delta \in \Delta_V$, we write $\gamma \vdash_{\overline{G}} \delta$ if either:

(1). there exist $(p, A\xi) \in \hat{\gamma}$ and a production $A \rightarrow X_1\eta_1X_2\eta_2\ldots X_k\eta_k$ in P such that $\delta = \gamma \cup \bigcup_{j=1}^{k} \{(p \cdot j, X_j\mu_j)\}$ where, for $1 \leqq j \leqq k$, $\mu_j = \eta_j\xi$ if $X_j \in N$, or $\mu_j = \varepsilon$ if $X_j \in T$, but $\delta = \gamma \cup \{(p \cdot 1, \varepsilon)\}$ when $k = 0$ holds (namely $A \rightarrow \varepsilon \in P$);

or

---

3) We write $\gamma \vdash^{*}_{\overline{G}} \delta$ if and only if either $\gamma = \delta$ or there exist $\gamma_0, \ldots, \gamma_n$ such that $\gamma_0 = \gamma$, $\gamma_i \vdash_{\overline{G}} \gamma_{i+1} (0 \leqq i \leqq n-1)$ and $\gamma_n = \delta$.

4) We use the following symbolic conventions, unless otherwise stated: Nonterminals: $A, B \in N$; Terminal: $a \in T$; Terminal or Nonterminal: $\chi \in N \cup T$; Index: $f \in F$; Strings of Indexes: $\eta, \xi, \mu \in F^*$; Positive Integers: $i, j \in J$; Strings of Positive Integers: $p, q, r \in J^*$.

(2). there exist $(p, Af\xi)\in\hat{\gamma}$ and an index production $A\rightarrow X_1X_2\cdots X_k$ in the index f such that $\delta=\gamma\cup\bigcup_{j=1}^{k}\{(p\cdot j,\ X_j\mu_j)\}$ where, for $1\leq j\leq k$, $\mu_j=\xi$ if $X_j\in N$, or $\mu_j=\varepsilon$ if $X_j\in T$, but $\delta=\gamma\cup\{(p\cdot 1,\ \varepsilon\}$ when $k=0$ holds (namely $A\rightarrow\varepsilon\in f$).

In the case 2 the index f is said to be *consumed* by the nonterminal A. Let $\left|\frac{*}{G}\right.$ be the reflexive and transitive closure of $\left|\frac{}{G}\right.$. Now we can formally define important subsets of Δ_V. We set

$\mathscr{T}(G)=\{\gamma\in\Delta_V\,|\,\{(0,\ S)\}\left|\frac{*}{G}\right.\ \gamma,\ g(\gamma)\in T^*\}$,

which is called the *set of terminal derivation trees of G*. Note that $L(G)=g(\mathscr{T}(G))$ is the language generated by G. We also set

$\mathscr{D}(G)=\{\gamma\in\Delta_V\,|\,\gamma=\Lambda,\ \text{or}\ \{(0,\ A\eta)\}\left|\frac{*}{G}\right.\ \gamma\}$,

which is called the *set ofderivation trees of G*, where Λ is a map from ϕ into V, called the *empty tree*. Clearly $\mathscr{T}(G)\subset\mathscr{D}(G)$ holds.

Here we notice that for any derivation tree $\gamma\in\mathscr{T}(G)$ and a node $p\in D_\gamma$, there holds exclusively $\gamma(p)=A\eta\in NF^*$ or $\gamma(p)=a\in T\cup\{\varepsilon\}$.

D. Definitions concerning the Trees of an Indexed Grammar

In the following we assume that an indexed grammar $G=(N,\ T,\ F,\ P,\ S)$ is given.

Definition 1.10. For $\gamma\in\Delta_V$ and $p\in D_\gamma$, we define the functions π_{1_γ}: $D_\gamma\rightarrow N\cup T\cup\{\varepsilon\}$, π_{2_γ}: $D_\gamma\rightarrow F^*$ and $\tilde{\pi}_{2_\gamma}$: $D_\gamma\rightarrow F\cup\{\varepsilon\}$ by setting

(a) $\pi_{1_\gamma}(p)=\begin{cases} X & \text{if } \gamma(p)=X\eta, \\ \varepsilon & \text{if } \gamma(p)=\varepsilon, \end{cases}$

(b) $\pi_{2_\gamma}(p)=\begin{cases} \eta & \text{if } \gamma(p)=X\eta, \\ \varepsilon & \text{if } \gamma(p)=\varepsilon, \end{cases}$

(c) $\tilde{\pi}_{2_\gamma}(p)=\begin{cases} f & \text{if } \pi_{2_\gamma}(p)=f\eta, \\ \varepsilon & \text{if } \pi_{2_\gamma}(p)=\varepsilon, \end{cases}$ where $X\in N\cup T$, $\eta\in F^*$ and $f\in F$.

The functions π_{1_γ} and π_{2_γ} can be considered as the projections of γ, and $\tilde{\pi}_{2_\gamma}$ is the function which picks up the leftmost index of $\pi_{2_\gamma}(p)$. $\pi_{2_\gamma}(p)$ is called the *indexed part* of a node p in D_γ. Hereafter the subscript γ of these functions is dropped whenever γ is clearly understood. Next we define relations $\leqq$ and $\lessdot$ on F^* which are convinient to describe the situation of consuming indexes.

Definition 1.11. For $\mu, \eta \in F^*$,

(a) $\eta \leqq \mu$ means that there exists ξ in F^* such that $\mu = \xi\eta$;

(b) $\eta \lessdot \mu$ means that $\eta \leqq \mu$ and $\eta \neq \mu$.

Now we define operations on the indexed part (i.e. $\pi_2(p)$), where there needs to note that $(p, a\mu)$ in $\gamma \in \mathscr{D}(G)$ implies $\mu = \varepsilon$, when a is in $T \cup \{\varepsilon\}$.

Definition 1.12. First we define the operation to place indices η under the indexed parts of a derivation tree $\gamma \in \mathscr{D}(G)$ by setting [5)]

$$\gamma \circ \eta = \{(p, X\xi) \mid (p, X\mu) \in \gamma, \text{ where } \xi = \varepsilon \text{ if } X \in T \cup \{\varepsilon\},$$
$$\text{or } \xi = \mu\eta \text{ if } X \in N\}.$$

To erase indices η from the indexed parts of a derivation tree $\gamma \in \mathscr{D}(G)$, there needs to hold the condition that for each $p \in D_\gamma$, we have $\eta \leqq \pi_2(p)$ or $\pi_1(p) \in T \cup \{\varepsilon\}$; for such γ and η we set

$$\gamma/\eta = \{(p, X\xi) \mid (p, X\mu) \in \gamma, \text{ where } \xi = \varepsilon \text{ if } X \in T \cup \{\varepsilon\},$$
$$\text{otherwise } \xi \in F^* \text{ such that } \mu = \xi\eta\}.$$

Next we introduce a fundamental notion which plays an important role in our proof of our main theorem.

Definition 1.13. For $\gamma \in \mathscr{D}(G)$, a function $e_\gamma \colon D_\gamma \to 2^{D_\gamma}$, called the *end of scope function*, is defined as follows:

$$e_\gamma(p) = \{q \in D_\gamma \mid p < q,\ \pi_2(p) = \pi_2(q),\ \pi_2(p) \underline{\lessdot} \pi_2(r)$$
$$\text{for any } r \in D_\gamma \text{ such that } p \leq r \leq q, \text{ and } \pi_2(q \cdot j) \lessdot$$
$$\pi_2(p) \text{ for any } j \in J \text{ such that } q \cdot j \in D_\gamma\}.$$

Intuitively, $e_\gamma(p)$ is the set of those nodes which are descendants of

5) Hereafter, we use the term *indexes* instead of saying a *string of indexes.*

p and occupy the places where the leftmost index of $\pi_2(p)$ is erased for the first time. (Of course, if no such nodes exist, $e_\gamma(p)$ is the empty set.)

The subset $scope_\gamma(p)$ of D_γ whose elements are the nodes from p to an element of $e_\gamma(p)$ is called the *scope of p*. Namely $scope_\gamma(p)=\{r\in D_\gamma \mid p<r \text{ and} q\nless r \text{ for any } q\in e_\gamma(p)\}$.

Remark 1.2. If $e_\gamma(p)\neq\phi$, then by the defintion of $e_\gamma(p)$, we can well-order the elements of $e_\gamma(p)$ by $\lessdot$.

The following function is also important.

Definition 1.14. A function n_γ: $D\to 2^N$ is defined as follows:

$$n_\gamma(p)=\{\pi_1(q)\in N \mid q\in e_\gamma(p)\},$$

where $\gamma\in\mathscr{D}(G)$ and N is the nonterminal alphabet of G.

$n_\gamma(p)$ is the set of nonterminals which label the nodes in $e_\gamma(p)$. Finally we define the concatenation of special trees as follows. This operation is used to express our main theorem more briefly.

Definition 1.15. For $\gamma, \delta\in\mathscr{D}(G)$ such that $g(\gamma)\in T^*A\eta T^*$ and $\delta=\Lambda$ or $\delta(0)=A\eta$, $\gamma\cdot\delta\in\mathscr{D}(G)$ is defined as follows:

$$\gamma\cdot\delta=\begin{cases}\gamma & \text{if } \delta=\Lambda, \text{ where } \Lambda \text{ is the empty tree,}\\ \gamma\cup p\cdot\delta & \text{if } \delta(0)=A\eta, \text{ where } (p, A\eta)\in\hat{\gamma}.\end{cases}$$

2. Decomposition of Derivation Trees

The aim of this section is to determine a constant k depending on a given indexed grammar G with the following property. If the number of maximal nodes of a terminal derivation tree γ in $\mathscr{T}(G)$ is greater than k, then there is a decomposition of γ which assures the existence of parts of γ that can be intercalated.

Definition 2.1. For each indexed grammar $G=(N, T, F, P, S)$, we define $r(G)$, the *rank of G*, by setting

$$r(G)=\max\{|\alpha| \mid A\to\alpha \text{ is a rule in } P\cup\bigcup_{f\in F} f\}.^{6)}$$

6) $|x|$ denotes the length of x.

If $r(G) \leqq 1$, then each production in P has one of the following forms

$$A \to B,\ A \to a,\ A \to \varepsilon.$$

Therefore such a grammar G is context-free and the language generated by G is finite. For such a simple grammar, our extended uvwxy-theorem which we will develop hereafter will hold trivially. Thus in the following we may assume without loss of generality that the rank of a given indexed grammar $G=(N, T, F, P, S)$ is greater than or equal to two.

Definition 2.2. For $\gamma \in \mathscr{D}(G)$, a node $p \in D_\gamma$ is called a *productive node* (abbreviated as *P-node*), if its *branching number*, $\max\{i \mid p \cdot i \in D_\gamma, i \in J\}$, is greater than or equal to two. This is the node which increases at least one element in the front $\hat{\gamma}$.

Definition 2.3. For $\gamma \in \mathscr{D}(G)$, a pair (p_0, p_1) of distinct nodes in D_γ is said to be *CF-like* if p_0 and p_1 satisfy the following conditions:

(1) $p_0 < p_1$.

(2) There exists at least one P-node p such that $p_0 \leqq p < p_1$.

(3) $\gamma(p_0) = \gamma(p_1)$.

Lemma 2.1. *For a terminal derivation tree $\gamma \in \mathscr{T}(G)$, if there exists a CF-like pair (p_0, p_1) of nodes in D_γ, then there exists a decomposition $\gamma = \alpha \cdot \beta \cdot \delta$ satisfying the conditions* (i) *and* (ii):

(i) *For each* $n \geqq 0$, $\gamma_n = \alpha \cdot \underbrace{\beta \cdot \beta \cdot \cdots \cdot \beta}_{n \text{ times}} \cdot \delta \in \mathscr{T}(G)$.

(ii) *For each* $n \geqq 1$, $\#(\hat{\gamma}_n) < \#(\hat{\gamma}_{n+1}) < (n+1)\#(\hat{\gamma})$.[7)]

Proof. Let $\alpha = \gamma \backslash p_0$, $\beta = \gamma < p_0, p_1 >$ and $\delta = \gamma / p_1$. Since the condition (3) of Definition 2.3 implies $\gamma(p_0) = \gamma(p_1) = A\eta$ (say), we have $\beta(0) = A\eta$ and $g(\beta) \in T^* A\eta T^*$. Therefore we have $\beta^n(0) = A\eta$ and $g(\beta^n) \in T^* A\eta T^*$ for each $n \geqq 1$, where $\beta^n = \underbrace{\beta \cdot \beta \cdot \cdots \cdot \beta}_{n \text{ times}}$. The facts $g(\alpha) \in T^* A\eta T^*$ and $\delta(0) = A\eta$ assure that $\gamma_0 = \alpha \cdot \delta \in \mathscr{T}(G)$ and $\gamma_n = \alpha \cdot \beta^n \cdot \delta \in \mathscr{T}(G)$ for each $n \geqq 1$. Thus the condition (i) is verified. By the condition (2) of Definition 2.3, a P-node exists in β. Therefore we have $2 \leqq \#(\hat{\beta}) \leqq \#(\hat{\gamma})$. On the

---

7) If A is a set, $\#(A)$ denotes the cardinality of A.

other hand, for each $n \geqq 0$, we have $\#(\hat{\gamma}_{n+1}) - \#(\hat{\gamma}_n) = \#(\hat{\beta}) - 1$. Thus the condition (ii) is verified.

Definition 2.4. A derivation tree $\gamma \in \mathscr{D}(G)$ is called a *non-CF-like tree* if γ has no CF-like pairs of nodes in D_γ.

What we want is a decomposition of a non CF-like terminal derivation tree $\gamma \in \mathscr{T}(G)$ which assures the existence of parts of γ that can be intercalated. For this purpose there needs to verify the existence of two nodes p_0 and p_1 in a chain C of γ such that $\pi_1(p_0) = \pi_1(p_1)$, $\tilde{\pi}_2(p_0) = \tilde{\pi}_2(p_1)$, $\pi_1(\bar{p}_0) = \pi_1(\bar{p}_1)$, $n_\gamma(p_0) = n_\gamma(p_1)$ and $scope_\gamma(p_1) \subset scope_\gamma(p_0)$, where $\bar{p}_i$ is the node of $e_\gamma(p_i)$(e_γ is the end of scope function) which is also in $C(i=0,1)$. Fig. 1 is an illustration. We will show that if $\#(\hat{\gamma}) > k$, then there exist such nodes p_0 and p_1 satisfying a bit more conditions, where k is a constant depending on G. We examine the details afterward.

Proposition 2.1. *For $\gamma \in \mathscr{D}(G)$, if $\#(\hat{\gamma}) > m^{k_1 - 1}$, then there exists a chain C of γ which contains at least k_1 P-nodes, where $m = r(G)$.*

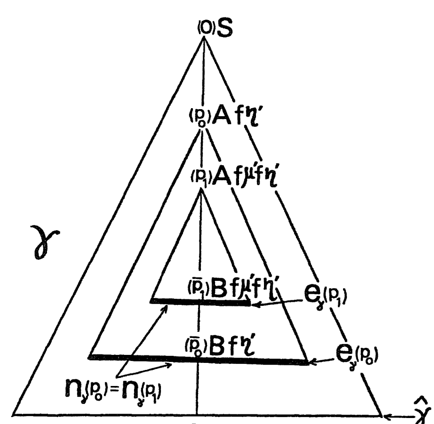


Fig. 1.

Proof. Since the branching number of any node in D_γ is less than or equal to m, if each chain of γ contained at most $k_1 - 1$ P-nodes, we would have $\#(\hat{\gamma}) \leqq m^{k_1 - 1}$. Therefore, if $\#(\hat{\gamma}) > m^{k_1 - 1}$, then there exists a chain C of γ containing k_1 P-nodes at least.

Definition 2.5. We set $\mathscr{A}(G) = \{\gamma \in \mathscr{D}(G) \mid \gamma(0) \in N,\ g(\gamma) \in (N \cup T)^*\}$.

Namely $\mathscr{A}(G)$ is the set of derivation trees each of which has only a nonterminal as its root label and a nonterminal, a terminal or ε as the label of each maximal node. Evidently we have $\mathscr{T}(G)\subseteqq\mathscr{A}(G)\subset\mathscr{D}(G)$. It will be clarified later in the proof of Lemma 2.2 why we introduce the new set $\mathscr{A}(G)$.

Definition 2.6. For a tree $\gamma\in\mathscr{A}(G)$ and for a chain

$$C=<(q_0,\ X_0\eta_0),\dots,(q_i,\ X_i\eta_i),\dots,(q_m,\ X_m\eta_m)>$$

of γ, we set

$$D_C=\{q_i\,|\,(q_i,\ X_i\eta_i)\in C\}-\{q_m\},$$

and

$$M_C=\{q\in D_C\,|\,\pi_{2\gamma}(q)=\varepsilon\}.$$

Let $q_{i_0}, q_{i_1},\dots, q_{i_n}$ be the elements of M_C where $i_0<i_1<\cdots<i_n$. Now we define a *moutain* M_j of C by setting

$$M_j=\{q\in D_C\,|\,q_{i_j}\leqq q<q_{i_{j+1}}\},\ (0\leqq j<n),$$

and

$$M_n=\{q\in D_C\,|\,q_{i_n}\leqq q\}$$

The mountain which contains at least one P-node is called a *productive mountain* (abbreviated as *P-mountain*).

Proposition 2.2. *If a chain C of a non-CF-like tree $\gamma\in\mathscr{A}(G)$ has more than vk_2 P-nodes, then there exists a P-mountain M which contains at least k_2 P-nodes where $v=\sharp(N)$.*

Proof. If there exist $v+1$ P-mountains of C, then there exist q_{j_0} and q_{j_1} in M_C such that $\pi_1(q_{j_0})=\pi_1(q_{j_1})\in N$, and $\pi_2(q_{j_0})=\pi_2(q_{j_1})=\varepsilon$. Therefore we have $\gamma(q_{j_0})=\gamma(q_{j_1})$, and since M_{j_0} is a P-mountain, $(q_{j_0},\ q_{j_1})$ is a CF-like pair. This leads to a contradiction. Thus γ has at most v P-mountains. Since $\gamma\in\mathscr{A}(G)$, we have $\pi_{2_\gamma}(0)=\varepsilon$, therefore each P-node belongs to some P-mountain. Thus if M is one of the P-mountains which contain the largest number of P-nodes, then there exist at least

k_2 P-nodes in M.

Definition 2.7. For a node q in a mountain M of a chain C of $\gamma \in \mathscr{A}(G)$, we set $\bar{e}(q)=M \cap e_\gamma(q)$, where e_γ is the end of scope function. $\bar{e}(q)$ is the node in M which occupies the place where the leftmost index of $\pi_2(q)$ is erased. Therefore $\bar{e}(q)$ is a singleton set or ϕ (this arises either when q is the first element of M or when M is the last mountain of C and all the indexes of $\pi_2(q)$ are not erased.).[8] Next we define a function ls: $M \rightarrow 2^M$ by setting

$$ls(q)=\{r \in M \mid q \leqq r \text{ and } r \leqq \bar{e}(q) \text{ if } \bar{e}(q) \neq \phi\}.$$

The set $ls(q)$ of nodes is called a *local scope of* q. Notice that we have $\pi_{2_\gamma}(q) \leqq \pi_{2_\gamma}(r)$ for any $r \in ls(q)$. If $\sharp(ls(q))=1$ then q is called a *descending* node. Note that $\bar{e}(q)$ equals $\{q\}$ or q is the last node of M if q is a descending node.

Let $'$ be the successor function for elements in M concerning the relation $<$. Namely for each $q \in M$, $q'=q \cdot j \in M$ for some $j \in J$. We can form nodes in M into a tree γ_M, considering whether q' is in $ls(q)$ or not. Using this tree, we can prove the existence of nodes in M which expand indexes and also increase P-nodes, provided that M contains enough P-nodes. Now we begin to work formally by constructing the following function *maketree*.

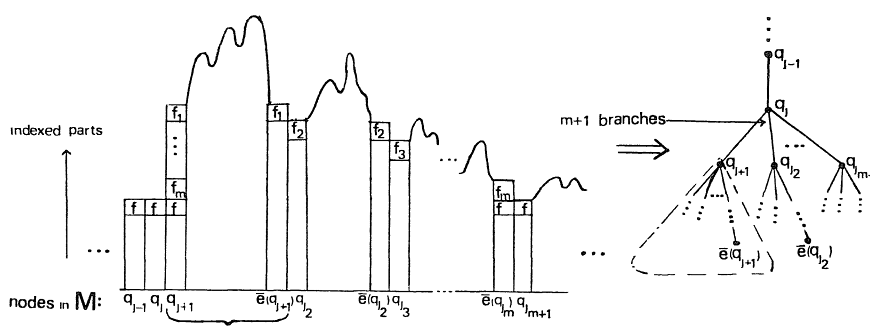
Definition 2.8. For $p \in J^*$ and $q \in M$, we set

$$maketree<p, ls(q)> = \begin{cases} \{(p, q)\} \text{ if } \sharp(ls(q))=1 \cdots\cdots & \text{(i)} \\ \{(p, q)\} \cup maketree<p\cdot 1, ls(q')> & \\ \qquad \text{if } \sharp(ls(q) \geqq 2 \text{ and} & \\ \qquad \pi_{2_\gamma}(q)=\pi_{2_\gamma}(q') \cdots\cdots & \text{(ii)} \\ \{(p, q)\} \cup \bigcup_{i=1}^{m_0} maketree<p\cdot i, ls(q_i)> & \\ \qquad \text{if } \sharp(ls(q)) \geqq 2 \text{ and} & \\ \qquad \pi_{2_\gamma}(q') = f_1 \cdots f_m \pi_{2_\gamma}(q) \cdots\cdots & \text{(iii)} \end{cases}$$

8) We write $\bar{e}(q)=p$ instead of $\bar{e}(q)=\{p\}$ when $\bar{e}(q) \neq \phi$.

The case (i) occurs when q is a descending node. The case (ii) occurs when q' is in $ls(q)$ and indices don't change. The case (iii) occurs when q' is in $ls(q)$ and indices are expanded. In the case (iii), m_0 and q_i are determined as follows. Beginning with setting $q_i = q'$, we want to let $q_{i+1} = \hat{q}'_i$ where $\bar{e}(q_i) = \hat{q}_i$ for $i = 1, 2, \ldots$. q_{i+1} is defined so long as $\bar{e}(q_i)$ is nonempty, assuring that the index f_i is consumed later, and is not a singleton containing the final element of M. If there appears some i such that $\bar{e}(q_i)$ is empty, or is a singleton containing the final element of M, then let m_0 be that i. Otherwise let $m_0 = m+1$. Note that each index f_i will be consumed at $\bar{e}(q_i)$ so that the expanded indices $f_1 \cdots f_m$ will be consumed completely at $\bar{e}(q_m)$ when $m_0 = m+1$. See Fig. 2. Now we define the desired tree γ_M by setting

$\gamma_M = maketree <0, ls(r)>$, where r is the first element of M.



This part corresponds to the dotted subtree.

The maketree function

Fig. 2.

Remark 2.1. γ_M has the following properties derived from its construction.

(a) For any $(p, q) \in \gamma_M$, we have $s \in ls(q)$ if and only if $(r, s) \in \gamma_M/p$ for some $r \in J^*$.

(b) Subtrees $\gamma_M/p \cdot i$ and $\gamma_M/p \cdot j$ where $i \neq j$ correspond to the disjoint intervals of the mountain M of γ as shown in Fig. 2. In addition the interval corresponding to $\gamma_M/p \cdot i$ appears first if $i < j$.

In the following definition, a node p of D_{γ_M} is called a MP-node if, for $(p, q) \in \gamma_M$, q is a P-node of $\gamma \in \mathscr{A}(G)$.

Definition 2.9. A node q in a P-mountain M of $\gamma\in\mathscr{A}(G)$ is called a *branching node* (*B-node*) if one of the following conditions is satisfied, where p is the unique node in D_{γ_M} such that $(p, q)\in\gamma_M$.

(i) q is a P-node of γ and there exists $j\in J$ such that $\gamma_M/p\cdot j$ has at least one MP-node.

(ii) There exist at least two distinct j_1 and j_2 in J such that $\gamma_M/p\cdot j_i$ has at least one MP-node ($i=1, 2$).

For $(p, q)\in\gamma_M$, if q is a B-node, then the node p in D_{γ_M} is called a *MB-node*.

Proposition 2.3. *For the tree γ_M, the followings hold:*

(1) *The youngest ancestor common to the two incomparable MP-nodes is an MB-node.*

(2) *If two comparable nodes are MP-nodes, then the one being the ancestor of the other is an MB-node.*

(3) *The youngest ancestor common to two incomparable MB-nodes is an MB-node.*

(4) *The totality of MB-nodes of γ_M forms a tree with respect to the ancestor-decendant relation induced by γ_M.*

(5) *For $(p, q)\in\gamma_M$, if q is a B-node of γ, then q is not a decending node, and for any other B-node $r\in ls(q)$, there exists at least one P-node in $ls(q)-ls(r)$.*

Proof. The conditions (1), (2) and (3) are immediate consequences of Definition 2.9. The condition (3) implies (4). Definition 2.9 and the condition (a) of Remark 2.1 assure (5).

Proposition 2.4. *If a P-mountain M of a chain C of $\gamma\in\mathscr{A}(G)$ has more than k_2 P-nodes, then there exists a chain $\bar{C}$ of γ_M which contains at least k_3 MB-nodes, where*

$$k_2=\frac{(s+1)^{k_3+1}-1}{s} \text{ and } s=\max\ \{|\eta_i|\mid A\rightarrow X_1\eta_1X_2\eta_2\dots X_k\eta_k\in P\}.$$

Proof. Using the condition (4) of Proposition 2.3, we let $\tilde{\gamma}_M$ be the tree consisting of MB-nodes. Since s indices may be expanded at a time, γ_M may have at most $s+1$ branches. Therefore $\tilde{\gamma}_M$ has also at most $s+1$

branches, because it preserves the ancestor-decendant relation of γ_M. If the length of any chain of $\tilde{\gamma}_M$ were less than k_3, then γ_M would have less than k_2 MP-nodes, since a MB-node may be a MP-node and $k_2=1+(s+1)+(s+1)^2+\cdots+(s+1)^{k_3}$. But γ_M has more than k_2 MP-nodes, therefore there exists a chain of $\tilde{\gamma}_M$ whose length is greater than or equal to k_3. We put this chain back on the original tree γ_M, and let $\bar{C}$ be the returned chain, which is the desired one.

Definition 2.10. For a P-mountain M of a chain C of $\gamma\in\mathscr{A}(G)$, we define a function $h:M\rightarrow N\times(F\cup\{\varepsilon\})\times(N\subset\{\$\})\times 2^N$ by setting

$$h(q)=\begin{cases}(\pi_1(q),\ \tilde{\pi}_2(q),\ \pi_1(\bar{e}(q)),\ n_\gamma(q)) & \text{if} \quad \bar{e}(q)\neq\phi,\\ (\pi_1(q),\ \tilde{\pi}_2(q),\quad \$ \quad,\ n_\gamma(q)) & \text{if} \quad \bar{e}(q)=\phi,\end{cases}$$

where \$ is a new symbol. Namely, for each node $q\in M$, the value $h(q)$ consists of the nonterminal symbol labelling it, the top index of its indexed part, the nonterminal symbol of $\bar{e}(q)$ (if it exists), and a set of nonterminals labelling the nodes in $e_\gamma(q)$. This function is called the *h-function.*

Lemma 2.2. *Given an indexed grammar $G=(N, T, F, P, S)$, there exists an integer k with the following property. For any non-CF-like terminal derivation tree $\gamma\in\mathscr{T}(G)$ such that $\#(\hat{\gamma})$ is greater than k, there exists a chain C of γ in which there exist two nodes p_0 and p_1 satisfying the following conditions:*

(i) *$p_0<p_1$ and both are B-nodes.*
(ii) *$h(p_0)=h(p_1)$.*
(iii) *$p_1\in ls(p_0)$.*
(iv) *$\pi_2(p_0)\lessdot\pi_2(p_1)$.*
(v) *There exists at least one P-node r in $ls(p_0)-ls(p_1)$.*
(vi) *$\#(\hat{\gamma}')\leqq k$, where $\gamma'=<p_0, e_\gamma(p_0)>/\pi_2(p_0)$.*

Proof. We note that the h-function of Definition 2.10 has at most $v(v+1)(t+1)2^v$ distinct values, where $v=\#(N)$ and $t=\#(F)$. We set a constant $k_3=2v(v+1)(t+1)2^v+1$, and let the constant k of this lemma be equal to m^{k_1-1}, where $m=r(G)$, $k_1=vk_2$ and $k_2=\dfrac{(s+1)^{k_3+1}-1}{s}$ ($s=\max\{|\eta_i|\mid A\rightarrow X_1\eta_1X_2\eta_2\ldots X_t\eta_t\in P)$.

Let C be one of the chains which contain the maximum number of P-nodes. By Proposition 2.1, C has at least k_1 P-nodes. From those P-mountains of C which contain the largest number of, we select the P-mountain M which is the nearest one from the front. M has at least k_2 P-nodes (Proposition 2.2). Next we apply Proposition 2.4 to this M, and let $\bar{C}$ be one of the chains of γ_M which contain the muximum number of MB-nodes, then $\bar{C}$ has at least k_3 MB-nodes. We set

$$M'=\{q\in M|(p, q)\in\bar{C} \text{ and } p \text{ is a } MB\text{-node.}\},$$

Then $\sharp(M')\geq k_3=2v(v+1)(t+1)2^v+1$ and M' is a set of B-nodes of γ. We can order the nodes in M' by $<$. In the k_3 B-nodes in M' counted from the maximal node of M', there exist three B-nodes p_{-1}, p_0 and p_1 whose h-function values are the same $(p_{-1}<p_0<p_1)$.

These p_0 and p_1 satisfy the conditions of this lemma except (vi). Evidently conditions (i) and (ii) are satisfied. The condition (iii) follows from (a) of Remark 2.1 and (v) follows from the condition (5) of Proposition 2.3. As for the condition (iv), since $\pi_2(p_0)\leqq\pi_2(p_1)$ already holds (because $p_1\in ls(p_0)$), we show that $\pi_2(p_0)=\pi_2(p_1)$ does not occur. Since p_0 and p_1 are B-nodes, if we had $\pi_2(p_0)=\pi_2(p_1)$, then there would exist a P-node q such that $p_0\leqq q<p_1$ because of the condition (b) of Remark 2.1 and (5) of Proposition 2.3. Therefore a pair (p_0, p_1) would become a CF-like one. This is a contradiction.

Now we examine the condition (vi). Since p_{-1}, p_0 and p_1 are selected among k_3 B-nodes in M' counted from the maximal node of M', and they are located on the chain $\bar{C}$ of γ_M which contains the maximum number of MB-nodes, γ_M/q contains at most k_2 MP-nodes, where $q\in J^*$ such that $(q, p_{-1})\in\gamma_M$. This means that there are at most k_2 P-nodes in $ls(p_{-1})$ (Remark 2.1). Since p_{-1} is a B-node and a B-node p_0 is in $ls(p_{-1})$, there exists a P-node in $ls(p_{-1})-ls(p_0)$ ((5) of Proposition 2.3). Therefore the part C' of C from p_0 to $\bar{e}(p_0)$ (or to the last node of C if $\bar{e}(p_0)=\phi$) corresponding to $ls(p_0)$ contains at most k_2-1 P-nodes. Since we have $h(p_{-1})=h(p_0)=h(p_1)$, there holds that $\bar{e}(p_{-1})$, $\bar{e}(p_0)$ and $\bar{e}(p_1)$ are simultaneously empty or nonempty. Thus two cases arise. We recall that γ' is equal to $\gamma<p_0, e_\gamma(p_0)>/\pi_2(p_0)$ in the condition (vi).

Case I: $\bar{e}(p_{-1})=\bar{e}(p_0)=\bar{e}(p_1)=\phi$

In this case C' becomes a chain of the subtree γ/p_0 containing most the P-nodes in γ/p_0 because C is so in γ. Since C' contains at most k_2-1 P-nodes, we have $\#(\hat{\alpha})\leqq m^{k_2-1}\leqq m^{k_1-1}=k$, where $\alpha=\gamma/p_0$. Since γ' is obtained from a part of α, we have $\#(\hat{\gamma}')\leqq\#(\hat{\alpha})\leqq k$. Thus the condition (vi) holds in this case.

Case II: $\bar{e}(p_{-1})\neq\phi$, $\bar{e}(p_0)\neq\phi$ and $\bar{e}(p_1)\neq\phi$

In this case though the part of C' of C from p_0 to $\bar{e}(p_0)$ contains at most k_2-1 P-nodes, it doesn't necessarily become a chain of γ' which has most the P-nodes in γ'. Therefore $\#(\hat{\gamma}')\leqq k$ is not always guaranted. If it holds, it is all right. We must consider the case that it doesn't hold.

Since γ is a non-CF-like tree, γ' is also a non-CF-like tree. In addition γ' is in $\mathscr{A}(G)$ because we have $\gamma'(0)\in N$ and $g(\gamma')\in(N\cup T)^*$. Therefore, if $\#(\hat{\gamma}')>k$ holds, Propositions 2.1~2.4 and related definitions are valid for γ'. Thus we can apply the procedure up to the present in this proof to γ' again. Now we obtain new p'_{-1}, p'_0 and p'_1, and do the case analysis. If Case I arises, it is all right. If Case II arises and the condition is not satisfied again, we repeat this process again.... This process halts without fail. Since there exists a P-node in $ls(p_{-1})-ls(p_0)$, $\#(\hat{\gamma}')<\#(\hat{\gamma})$ always holds. The cardinarity of the front is decreased every time the process is repeated. The B-node p_{-1} is used for this purpose. Thus the process eventually reaches the stage where either Case I or Case II with the desired situation arises and it halts.

When our process halts, we finally obtain the desired three nodes p_{-1}, p_0 and p_1 positioned in a chain C of γ after the renaming if necessary, and these p_0 and p_1 satisfy the conditions (i)~(vi).

3. Intercalation Lemma

Using Lemma 2.2 in the previous section, we prove the following lemma which throws light upon how terminal derivation trees increase.

Lemma 3.1. *For each indexed granmar G, there exists an integer k with the following property. For any non-CF-like tree $\gamma\in\mathscr{T}(G)$ such that $\#(\hat{\gamma})>k$, we can effectively construct α and ν and, for each $i\geqq 0$, β_{i+1}, δ_i and τ_{i+1} in $\mathscr{D}(G)$, and γ is decomposed into $\alpha\cdot\beta_1\cdot\delta_1\cdot\tau_1\cdot\nu$ and the follow-*

ing conditions (i)~(iii) *are satisfied.*

(1) $\gamma_0=\alpha\cdot\delta_0\cdot\nu\in\mathscr{T}(G)$ *and for* $n\geqq 1$
$\gamma_n=\alpha\cdot\beta_1\cdot\beta_2\cdot\cdots\cdot\beta_n\cdot\delta_n\cdot\tau_n\cdot\cdots\cdot\tau_2\cdot\tau_1\cdot\nu\in\mathscr{T}(G)$.

(2) *For each* $n\geqq 1$, $\#(\hat{\beta}_n)\leqq k_\gamma k^n$, $\#(\hat{\tau}_n)\leqq k_\gamma k^n$, $\#(\hat{\delta}_n)\leqq k_\gamma k^{n+1}$ *and* $\#(\hat{\gamma}_n)<3k_\gamma k^{n+1}$, *where* $k_\gamma=\#(\hat{\gamma})$.

(3) *For each* $n\geqq 1$, *either* β_n *or* τ_n *has at least one node whose label is a terminal or* ε.

Proof. The internal structure of α, β_{i+1}, δ_i, τ_{i+1} and ν is given in this proof. The constant k of this lemma is as in Lemma 2.2. Therefore we apply Lemma 2.2 to γ and get two nodes p_0 and p_1 satisfying the conditions of Lemma 2.2. We use the notations $\eta=\pi_2(p_0)$ and $\mu\eta=\pi_2(p_1)$ in the following (cf. the condition iv).[9] Note that the leftmost index of η, say f, is the same as that of μ (the conditions ii and iv). Therefore μ and η can be rewritten as $f\mu'$ and $f\eta'$ respectively.

Case I: $n_\gamma(p_0)=n_\gamma(p_1)=\phi$, (namely $e_\gamma(p_0)=e_\gamma(p_1)=\phi$).

This means that the string of indexes μ isn't consumed. Therefore there exists a P-node in $\gamma<p_0, p_1>$ which increases a terminal or ε (the condition v), so that this case is similar to Lemma 2.1 except for the treatment of indexes. We set

$\alpha=\gamma/p_0$, for each $i\geqq 1$, $\beta_i=\beta\circ\mu^{i-1}\eta$, where $\beta=\gamma<p_0, p_1>/\eta$, for each $i\geqq 0$, $\delta_i=\delta\circ\mu^i\eta$, where $\delta=(\gamma/p_1)/\mu\eta$, for each $i\geqq 1$, $\tau_i=\Lambda$, and $\nu=\Lambda$, where Λ is the empty tree.

Then we have $\beta_i(0)=A\mu^{i-1}\eta$ and $g(\beta_i)=A\mu^i\eta$ for each $i\geqq 1$, where $A=\pi_1(p_0)=\pi_1(p_1)$ (the condition ii). the fact that $g(\alpha)\in T^*A\eta T^*$, $\delta_i(0)=A\mu^i\eta$ for each $i\geqq 0$, and indexes μ isn't consumed, implies the condition (1). Since β has a P-node, each β_i contains a P-node. Thus the condition (3) is verified. For each $i\geqq 1$, we have $\#(\hat{\beta}_i)=\#(\hat{\beta})<\#(\hat{\gamma})$ and $\#(\hat{\delta}_i)=\#(\hat{\delta})<\#(\hat{\gamma})$. Therefore we have $\#(\hat{\gamma}_{n+1})-\#(\hat{\gamma}_n)=\#(\hat{\beta})-1$ for each $n\geqq 0$. Thus we have $\#(\hat{\gamma}_n)<\#(\hat{\gamma}_{n+1})<(n+1)\#(\hat{\gamma})$ for each $n\geqq 1$. Since k is greater than 1, the condition (2) is verified.

---

9) In this proof, the conditions within parentheses are those of Lemma 2.2 which are used to verify the claim to which they are attached.

Case II: $n_\gamma(p_0)=n_\gamma(p_1)\neq\phi$.

This means that $e_\gamma(p_0)\neq\phi$ and $e_\gamma(p_1)\neq\phi$. Therefore for each $q\in e_\gamma(p_1)$, let $e_q=\{r\in e_\gamma(p_0)\,|\,q<r\}$, $e_{p_0}=e_\gamma(p_0)-\bigcup_{q\in e_\gamma(p_1)} e_q$, and $T_0=\{\gamma/q\,|\,q\in e_\gamma(p_0)\}$. See Fig, 3. The Case II is divided into two subcases.

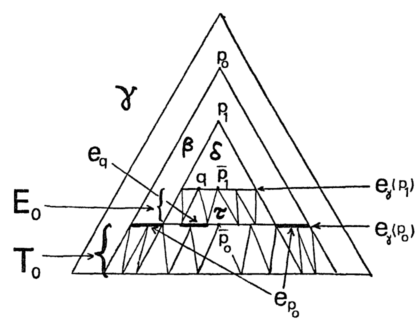


Fig. 3.

Case II-I: $\bar{e}(p_0)\neq\phi$ and $\bar{e}(p_1)\neq\phi$

Let $\bar{e}(p_0)=\bar{p}_0$, $\bar{e}(p_1)=\bar{p}_1$, $E_0=\{\gamma<q,\, e_q>/\eta\,|\,q\in e_\gamma(p_1)\}$, and for each $n\geqq 1$, $E_n=\{\tau\circ\mu^{n-1}\eta\,|\,\tau\in E_0\}$. Each tree ρ in T_0 consumes indexes η, because $\pi_{2_\rho}(0)=\eta$ and $g(\rho)\in T^*$. In addition the common leftmost index f of μ and η is consumed first. Namely we have $\pi_{2_\rho}(0)=f\eta'$ and $\pi_{2_\rho}(i)\leqq\eta'$ for each $i\in D_\rho$, where $\eta=f\eta'$. Each tree ρ in E_0 consumes indexes μ, because $\pi_{2_\rho}(0)=\eta$ and $g(\rho)\in(N\cup T)^*$. In addition we also have $\pi_{2_\rho}(0)=f\mu'$ and $\pi_{2_\rho}(i)\leqq\mu'$ for each $i\in D_\rho$, where $\mu=f\mu'$.

The condition $n_\gamma(p_0)=n_\gamma(p_1)$ (the condition ii) means that for each nonterminal labelling a maximal node of a tree in E_0, there always exists a tree in E_0 whose label of the root node is that nonterminal. Namely we have

$$\{A\in N\,|\,(p,\,A)\in\hat{\tau},\,\tau\in E_0\}\subseteq n_\gamma(p_0)=n_\gamma(p_1)=\{\pi_{1_\tau}(0)\,|\,\tau\in E_0\}.$$

This and the fact that the leftmost index of μ is identical to that of η assure that indexes $\mu^n\eta$ expanded on a tree in E_n can be consumed by the appropriately selected trees in $E_{n-1}, E_{n-2},\ldots, E_1$ and T_0 joined properly to it in this order. Namely for each $\rho_0\in E_n$ such that $g(\rho_0)\notin T^*$, we have $\rho_0(0)=A\mu^n\eta$ for some $A\in N$ and $g(\rho_0)\in T^*A_0\mu^{n-1}\eta T^*\ldots T^*A_m\mu^{n-1}\eta T^*$, and there exist $\sigma_0,\dots,\sigma_m \in E_{n-1}$ such that $\sigma_i(0)=A_i\mu^{n-1}(0\leqq i\leqq m)$. Then we get a new tree $\rho_1=\rho_0\begin{bmatrix} q_0,\dots, q_m \\ \sigma_0,\dots,\sigma_m\end{bmatrix}$, where $(q_i, A_i\mu^{n-1}\eta)\in\hat{\rho}_0$. Furthermore for ρ_1 with $g(\rho_1)\notin T^*$, we can insert proper trees in E_{n-2} into the places of the maximal nodes of ρ_1 whose labels are not terminals or ε. Thus we get a new tree ρ_2. Unless $g(\rho_i)\in T^*$, we continue this process and finally get a tree ρ_{n-1} the yield $g(\rho_{n-1})$ of which is in $T^*A'_0\eta T^*\dots T^*A'_{m'}\eta T^*$. Then there exist $t_0,\dots,t_{m'}\in T_0$ such that $t_i(0)=A'_i\eta(0\leqq i\leqq m')$, because we have $\{A\in N\,|\,(p, A)\in\hat{\tau},\ \hat{\tau}\in E_0\}\subseteq n_\gamma(p_0)=\{\pi_{1_\tau}(0)\,|\,\tau\in T_0\}$. Thus indexes η can be consumed, and we get a tree $\rho_n=\rho_{n-1}\begin{bmatrix} q'_0,\dots, q'_{m'} \\ t_0,\dots, t_{m'}\end{bmatrix}$, where $(q'_i, A'_i\eta)\in\hat{\rho}_{n-1}$, which is called a *complete* tree because $g(\rho_n)\in T^*$. If we have $g(\rho_i)\in T^*$ for some i such that $0\leqq i\leqq n-1$, we also call this ρ_i a *complete* tree. Note that for each $\rho\in E_n$ there may exist several complete trees depending on the choice of trees taken from $E_{n-1},\dots,E_1$ and T_0. For each $n\geqq 1$, let T_n be the set of all complete trees obtained from all the trees in E_n.

Now the desired decomposition is $\alpha=\gamma/p_0$, $\beta_1=\gamma<p_0, p_1>$, $\delta_1=\gamma<p_1, \bar{p}_1>$, $\tau_1=\gamma<\bar{p}_1, \bar{p}_0>$ and $\nu=\gamma/\bar{p}_0$. Next we must determine $\beta_i, \delta_i, \tau_i$ $(i\geqq 2)$ and δ_0. For this purpose, let $\beta=\gamma<p_0, p_1\cup e_{p_0}>/\eta$, $\delta=\gamma<p_1, e_\gamma(p_1)>/\mu\eta$ and $\tau=\gamma<\bar{p}_1, e_{\bar{p}_1}>/\eta$ (cf. Fig. 3), then β expands indexes μ, τ has nothing to do with μ, and τ which is an element in E_0 consumes μ. They have the following structures:

(1) $\beta(0)=A$ and $g(\beta)\in T^*A_1T^*\dots T^*A\mu T^*\dots T^*A_tT^*$, where $A=\pi_1(p_0)=\pi_1(p_1)$;

(2) $\delta(0)=A$ and $g(\delta)\in T^*A'_1T^*\dots T^*BT^*\dots T^*A'_mT^*$, where $B=\pi_1(\bar{p}_0)=\pi_1(\bar{p}_1)$;

(3) $\tau(0)=B\mu$ and $g(\tau)\in T^*A''_1T^*\dots T^*BT^*\dots T^*A''_nT^*$.

Next we set $\beta'_i=\beta\circ\mu^{i-1}\eta$, $\delta'_i=\delta\circ\mu^i\eta$ and $\tau'_i=\tau\circ\mu^{i-1}\eta$, for each $i\geqq 2$, then we have

(1′) $\beta'_i(0)=A\mu^{i-1}\eta$ and $g(\beta'_i)\in T^*A_1\mu^{i-1}\eta T^*\dots T^*A\mu^i\eta T^*\dots$
$\dots T^*A_t\mu^{i-1}\eta T^*$;

(2′) $\delta'_i(0)=A\mu^i\eta$ and $g(\delta'_i)\in T^*A'_1\mu^i\eta T^*\dots T^*B\mu^i\eta T^*\dots$
$\dots T^*A'_m\mu^i\eta T^*$;

(3′) $\tau'_i(0)=B\mu^i\eta$ and $g(\tau'_i)\in T^*A''_1\mu^{i-1}\eta T^*\dots T^*B\mu^{i-1}\eta T^*\dots$
$\dots T^*A''_1\mu^{i-1}\eta T^*$.

Since $\{A_1,\dots, A_t\}\subseteq n_\gamma(p_0)$, there exist $\rho_1,\dots, \rho_t\in T_{i-1}$ such that $\rho_j(0)=A_j\mu^{i-1}\eta(1\leqq j\leqq t)$. Thus we set $\beta_i=\beta'_i\begin{bmatrix} q_1,\dots, q_t \\ \rho_1,\dots, \rho_t\end{bmatrix}$, where $(q_j, A_j\mu^{i-1}\eta)\in \hat{\beta}'_i$, then we have $\beta_i(0)=A\mu^{i-1}\eta$ and $g(\beta_i)\in T^*A\mu^i\eta T^*$. Since $\{A'_1,\dots, A'_n\}\subseteqq n_\gamma(p_1)$, for δ'_i, there exist $\rho'_i,\dots, \rho'_m\in T_i$ such that $\rho'_j(0)=A'_j\mu^i\eta(1\leqq j\leqq m)$. Thus we set $\delta_i=\delta'_i\begin{bmatrix} q'_1,\dots, q'_m \\ \rho'_1,\dots, \rho'_m\end{bmatrix}$, where $(q'_j, A'_j\mu^i\eta)\in\hat{\delta}'_i$, then we have $\delta_i(0)=A\mu^i\eta$ and $g(\delta_i)\in T^*B\mu^i\eta T^*$. In the same manner, we can get τ_i such that $\tau_i(0)=B\mu^i\eta$ and $g(\tau_i)\in T^*B\mu^{i-1}\eta T^*$, by inserting proper trees in T_{i-1} because $\{A''_1,\dots, A''_n\}\subseteqq n_\gamma(p_0)$.

By the above construction, we can set $\gamma_n=\alpha\cdot\beta_1\cdot\beta_2\cdot\dots\cdot\beta_n\cdot\delta_n\cdot\tau_n\cdot\tau_{n-1}\cdot\dots\cdot\tau_1\cdot\nu\in\mathscr{T}(G)$, for $n\geqq 1$. Set $\delta_0=(\delta\circ\eta)\begin{bmatrix} q'_1,\dots,q'_m \\ \theta_1,\dots, \theta_m\end{bmatrix}$, where $\theta_j\in T_0$ such that $\theta_j(0)=A'_j\eta(1\leqq j\leqq m)$, then we also have $\gamma_0=\alpha\cdot\delta_0\cdot\nu\in\mathscr{T}(G)$, becaues $g(\alpha)\in T^*A\eta T^*$,$\delta_0(0)=A\eta$, $g(\delta_0)\in T^*B\eta T^*$ and $\nu(0)=B\eta$ hold. Thus condition (1) of this lemma is verified. Note that β_i, δ_i and τ_i have not been uniquely determined, since there may exist another proper trees in T_{i-1}, T_i and T_{i-1} which can be inlaid into β'_i, δ'_i and τ'_i respectively. But the following discussion holds for any choice.

Since p_0 is a B-node, there exists at least one P-node in β or τ (the condition v). Therefore either β_i or τ_i has at least one node whose label is a terminal or ε. Thus the condition (3) is satisfied. Consequently, if we can choose $\{\delta_n\}$ skillfully so that for $n\geqq 1$, $\#(\hat{\delta}_n)\leqq\#(\hat{\delta}_{n+1})$ may hold, then $\#(\hat{\gamma}_n)<\#(\hat{\gamma}_{n+1})$ holds for $n\geqq 1$. Concerning this point, we will discuss later at Lemma 4.1.

Now let's clarify the condition (2) of this lemma. First of all, since every tree in T_0 is a subtree of γ, we have

$$\max\{\#(\hat{\rho})\mid\rho\in T_0\}\geqq k_\gamma,\ \text{where}\ k_\gamma=\#(\hat{\gamma}).$$

Since every tree in E_0 is contained in $\gamma'=\gamma<p_0, e_\gamma(p_0)>/\eta$ and $\#(\hat{\gamma}')\leqq k$ (the condition vi), we have $\max\{\#(\hat{\rho})\mid\rho\in E_0\}\leqq k$. Therefore we have $\max\{\#(\hat{\rho})\mid\rho\in T_n\}\leqq\underbrace{k\cdots k}_{n\ \text{times}}\cdot k_\gamma=k_\gamma k^n$, because T_n is obtained through the use of $E_n, E_{n-1},\dots, E_1$ and T_0 successively. In the same manner since β, δ, τ are also obtained from γ', we have

$$\#(\hat{\beta}),\ \#(\hat{\delta}),\ \#(\hat{\tau})\leqq\#(\hat{\gamma}')\leqq k.$$

Since we make β_n, δ_n and τ_n of β, δ and τ by using proper trees in T_{n-1}, T_n and T_{n-1} respectively, there hold $\#(\hat{\beta}_n)\leqq k\cdot k_\gamma k^{n-1}=k_\gamma k^n$, $\#(\hat{\delta}_n)\leqq k_\gamma k^{n+1}$ and $\#(\hat{\tau}_n)\leqq k_\gamma k^n$. Since α and ν are subtrees of γ, we have $\#(\hat{\alpha})$, $\#(\hat{\nu})\leqq k_\gamma$. Therefore we get

$$\begin{aligned}\#(\hat{\gamma}_n)&=\#(\hat{\alpha})+\sum_{i=1}^{n}\#(\hat{\beta}_i)+\#(\hat{\delta}_n)+\sum_{i=1}^{n}\#(\hat{\tau}_i)+\#(\hat{\nu})-2(n+2)\\&\leqq 2k_\gamma+2\sum_{i=1}^{n}k_\gamma k^i+k_\gamma k^{n+1}-2(n+2)\\&<k_\gamma\left(2+2\frac{k^{n+1}-1}{k-1}+k^{n+1}\right)\leqq k_\gamma(2+2k^{n+1}-2+k^{n+1})\\&=3k_\gamma k^{n+1}.\end{aligned}$$

Thus the condition (2) is verified.

Case II-II: $\bar{e}(p_0)=\bar{e}(p_1)=\phi$.

This case can be treated in the same manner as Case II–I, except for the fact that $\tau_i=\Lambda(i\geqq 1)$, and $\nu=\Lambda$, where Λ is the empty tree.

4. Strict Growth of Fronts

Lemma 3.1 throws light upon how terminal derivation trees increase. But the strict growth of the fronts (*i.e.* $\#(\hat{\gamma}_n)<\#(\hat{\gamma}_{n+1})$ for $n\geqq 1$) is not necessarily guaranteed. We give an assurance by the following lemma. We notice that our proof depends heavily on the notations used in the proof of Lemma 3.1.

Lemma 4.1. *For each indexed grammar* $G=(N, T, F, P, S)$, *there exist integers k and c with the following property. For any non-CF-like tree* $\theta\in\mathscr{T}(G)$ *such that* $\#(\hat{\theta})>k$, *there exist trees* ρ, ω, σ_i, χ_i *and* $\psi_i\in\mathscr{D}(G)$ (*for each* $i\geqq 1$), *and* θ *is decomposed into* $\rho\cdot\sigma_1\cdot\chi_1\cdot\psi\cdot\omega$ *and the following conditions* (i) *and* (ii) *are satisfied*:

(i) *For each* $n\geqq 1$,
$\theta_n=\rho\cdot\sigma_1\cdot\sigma_2\cdot\cdots\cdot\sigma_n\cdot\chi_n\cdot\psi_n\cdot\cdots\cdot\psi_2\cdot\psi_1\cdot\omega\in\mathscr{T}(G)$

(ii) $\#(\hat{\theta}_n)<\#(\hat{\theta}_{n+1})<c_\theta c^{n+1}$ (*for each* $n\geqq 1$), *where* c_θ *is a constant depending on* θ.

Proof. The constant k is the same one as in Lemma 3.1. c and c_θ will be calculated to satisfy the lemma, using k and other constants. Applying Lemma 3.1 to θ, we get the decomposition $\theta=\alpha\cdot\beta\cdot_1\cdot\delta_1\cdot\tau_1\cdot\nu$. Therefore we set $\rho=\alpha$, $\sigma_1=\beta_1$, $\chi_1=\delta_1$, $\psi_1=\tau_1$ and $\omega=\nu$.

Next we look for σ_i, χ_i and $\psi_i(i\geqq 2)$ to satisfy this lemma, using Lemma 3.1. For the case I of Lemma 3.1, this lemma has already been satisfied, if we set $\sigma_i=\beta_i$, $\chi_i=\delta_i$ and $\psi_i=\tau_i(i>2)$. There holds $\#(\hat{\theta}_n)<\#(\hat{\theta}_{n+1})<(n+1)\#(\hat{\theta})<c_\theta c^{n+1}$ (for $n\geqq 1$) if we take the integers c and c_θ to be greater than $c>1$ and $\#(\hat{\theta})$ respectively. Explicit formulas for c and c_θ will be given later. It is sufficient to consider only the case II of Lemma 3.1. An outline of our construction is as follows.

Let $\{\gamma_n\}$ be a sequence of terminal derivation trees obtained from θ, using Lemma 3.1. Since γ_n is constructed by removing δ_{n-1} from γ_{n-1} and intercalating $\beta_n\cdot\delta_n\cdot\tau_n$ instead, the construction of γ_n from γ_{n-1} in this way is called *the n-th stage*. We pick up the v-th stage especially, where $v=\#(N)$, and give an algorithm to construct a sequence of trees $\{\delta'_j\}$ starting from $\delta_v(=\delta'_0)$. This is Step 1. Using this δ'_j instead of δ_{n_j} (where $n_j=jv!+v$) at the n_j-th stage, we will determine $\{\theta_n\}$ to satisfy the conditions of this lemma. This is Step 2.

Step 1: We construct $\{\delta'_j\}$ to satisfy the following conditions: For $j\geqq 0$,

(1) $\delta'_j(0)=\delta_{n_j}(0)$, where $n_j=jv!+v$;

(2) $g(\delta'_j)\in T^*B\mu^{n_j}\eta T^*$, where $B\in N$ such that $g(\delta_v)\in T^*B\mu^v\eta T^*$;

(3) $\#(\hat{\delta}'_j)\leqq\#(\hat{\delta}'_{j+1})$ and $\#(\hat{\delta}'_j)\leqq\#(\hat{\theta})k^{n_j+1}$.

We recall that $\delta'_0=\delta_v=(\delta\circ\mu^v\eta)\begin{bmatrix} q_1,\dots, & q_m \\ t_1,\dots, & t_m\end{bmatrix}$ in the case II, where $\delta=\theta<p_1,\ e_\theta(p_1)>/\mu\eta$, $\{p_1\cdot q_1,\dots,\ p_1\cdot q_m\}=e_\theta(p_1)-\{\bar{p}_1\}$ and $t_i=\delta_v/q_i\in T_v$ $(1\leqq i\leqq m)$. Starting from each t_i, we construct sequences of terminal trees $\{t_i^{(j)}\}(j\geqq 0, t_i^{(0)}=t_i, 1\leqq i\leqq m)$ such that $\#(\hat{t}_i^{(j)})\leqq\#(\hat{t}_i^{(j+1)})$. Using these $\{t_i^{(j)}\}$, we set $\delta'_j=(\delta\circ\mu^{n_j}\eta)\begin{bmatrix} q_1,\dots, & q_m \\ t_1^{(j)},\dots, & t_m^{(j)}\end{bmatrix}$. Then we have $\#(\hat{\delta}'_j)\leqq\#(\hat{\delta}'_{j+1})$. Therefore we must prove the following assertion

Assertion: *For each $t\in\{t_i,\dots t_m\}$, there exists a sequence of tree $\{t^{(j)}\}$ $(j\geqq 0,\ t^{(0)}=t)$ satisfying the following conditions. For $j\geqq 0$,*

(a) $t^{(j)}(0)=A\mu^{jv!+v}\eta$, $g(t^{(j)})\in T^*$ *and* $t^{(j)}\in T_{jv!+v}$ *where* $A=\pi_{1_t}(0)$
(b) $\#(\hat{t}^{(j)})\leqq\#(\hat{t}^{(j+1)})$.

Proof of the assertion. Since t is in T_v, t is formed by joining proper trees in $E_v, E_{v-1},\dots, E_1$ and T_0, and it consumes indexes $\mu^v\eta$. Therefore we set $K_v=\{0\}$ and $K_i=\{p\in D_t \mid (p, A\mu^i\eta)\in t,\ A\in N,\ p$ is used as a joint node to connect a tree in E_{i+1} with one in $E_i(T_0$ when $i=0).\}$ (for $0\leqq i\leqq v-1$). Two cases arise depending on whether $K_0=\phi$ or not.

Case I: $K_0\neq\phi$

We set $K=\bigcup_{0\leqq i\leqq v} K_i$, and each element in K is called a *M-node*. Noting $K_v=\{0\}$, we consider all paths from 0 to each element in K_0 and let K' be the set of all M-nodes on these paths. The subtree t/p of t at $p\in K-K'$ forms a terminal tree (i.e. $g(t/p)\in T^*$) without consuming indexes $\mu^v\eta$ completely. For each $p\in K'$, let $D_p=\{q\in K' \mid p<q\}$ and for each $i(0\leqq i\leqq v)$, let $K'_i=K_i\cap K'$.

We define a set $R\subset K'\times K'$ by using the following algorithm 1.

Algorithm 1: (1) Initially set $H:=\{0\}$ and $i:=v$, where H is a set variable.

(2) Set $H'=\{p\in H \mid$ there exists at least one node $q\in D_p$ such that $\pi_{1_t}(p)=\pi_{1_t}(q)\}$. For each $p\in H'$, select one such node $q\in D_p$, and let $[p, q]$ be an element of the set R.

(3) If $H=H'$, then halt. Otherwise reset $H:=\bigcup_{p\in H-H'}(D_p\cap K'_{i-1})$ and $i:=i-1$, go to (2).

The algorithm 1 always halts without becoming $i=0$, since $v+1$ elements of K' are located on each path from 0 to an element in K_0. The set R thus obtained has the following properties derived from its construction.

(i) If $[p, q]$ is in R, then $[r, q']$ is not in R for any $r\in K'$ satisfying $r<p$ and for any element $q'\in D_r$.

(ii) For each path C between 0 and an element in K_0, there exist a M-node p on C and $q\in D_p$ such that $[p, q]\in R$.

Depending on whether $R=\{[0, r]\}$ for some $r\in K'$ or not, two subcases arise. The former is the case I–I and the latter is the case I–II. (Clearly

R is nonempty.)

Case I-I: $R=\{[0, r]\}$ for some $r \in K'$.

We have $\pi_{1_t}(0)=\pi_{1_t}(r)=A$ (say) and $t(r)=A\mu^{v-s}\eta$, where s is an integer such that $s \leq v$. Namely indexes μ^s are consumed between 0 and r because $t(0)=A\mu^v\eta$. Thus there exists an integer h such that $v!=sh$. We set $u=(t\backslash K_{v-s})/\mu^{v-s}\eta$ and $u_i=u\circ\mu^{si+v}\eta$ (for each $i\geq 0$). Then we have that u consumes indexes μ^s, $u_i(0)=A\mu^{s(i+1)+v}\eta$ and $g(u_i)\in T^*A_1\mu^{si+v}\eta T^*\cdots T^*A\mu^{si+v}\eta T^*\cdots T^*A_m\mu^{si+v}\eta T^*$. We can change u_i into $\tilde{u}_i$ by attaching a proper tree in T_{si+v} to each node labelled $A_j\mu^{si+v}\eta$ so that $\tilde{u}_i(0)=A\mu^{s(i+1)+v}\eta$ and $g(\tilde{u}_i)\in T^*A\mu^{si+v}\eta T^*$. Since we have $g(\tilde{u}_0)\in T^*A\mu^v\eta T^*$ and $t(0)=A\mu^v\eta$, we can set $t^{(1)}=\tilde{u}_{h-1}\cdot\tilde{u}_{h-2}\cdot\cdots\cdot\tilde{u}_0\cdot t$ so that $t^{(1)}(0)=A\mu^{sh+v}\eta=A\mu^{v!+v}\eta$, $g(t^{(1)})\in T^*$ and $t^{(1)}\in T_{v!+v}$. In general we set $t^{(j)}=\tilde{u}_{hj-1}\cdot\tilde{u}_{hj-2}\cdot\cdots\tilde{u}_{h(j-1)}\cdot t^{(j-1)}(j\geq 1,\ t^{(0)}=t)$, then we have $t^{(j)}(0)=A\mu^{jv!+v}\eta$, $g(t^{(j)})\in T^*$ and $t^{(j)}\in T_{jv!+v}$. Thus the condition (a) of the assertion is satisfied. Since $t^{(j)}$ is embedded in $t^{(j+1)}$ whenever indexes $\mu^{v!}$ is increased, we have the condition (b). Thus $\{t^{(j)}\}$ is the desired sequence.

Case I-II: $R\neq\{[0, r]\}$ for any $r\in K'$.

Let $E=\{p\,|\,[p, q]\in R\}$. By the properties (i) and (ii) of R, t is decomposed into $t\backslash E$ and $\bigcup_{p\in E} t/p$, and the same situation as Case I–I arises for each t/p. The precise description is as follows.

For each $[p, q]\in R$, we have $\pi_{1_t}(p)=\pi_{1_t}(q)=B$(say), $t(p)=B\mu^s\eta$ and $t(q)=B\mu^{s-s'}\eta$, where s and s' are integers such that $1\leqq s'\leqq s<v$. These mean that indexes μ^{v-s} are consumed between 0 and p, and $\mu^{s'}$ are consumed between p and q. There exists an integer h' such that $v!=s'h'$. Therefore we can apply the construction method of Case I–I to t/p and we get a sequence of trees $\{\alpha^{(j)}\}$ such that $\alpha^{(0)}=t/p$, $\alpha^{(j)}(0)=B\mu^{jv!+s}\eta$, $g(\alpha^{(j)})\in T^*$ and $\alpha^{(j)}\in T_{jv!+s}$, for $j\geqq 0$. Note that $\alpha^{(j)}$ is embedded in $\alpha^{(j+1)}$ for $j\geqq 0$.

Now let $E=\{p_0,\ldots, p_n\}$, $t(0)=A\mu^v\eta$, $t(p_i)=B_i\mu^{s_i}\eta(0\leqq i\leqq n)$, and $w=(t\backslash E)/\eta$, then we have a sequence of trees $\{\alpha_i^{(j)}\}$ constructed by the above method for each t/p_i. Next we set $w_j=w\circ\mu^{jv!}\eta$, then we have $w_j(0)=A\mu^{jv!+v}\eta$ and $g(w_j)\in T^*B_0\mu^{jv!+s_0}\eta T^*\cdots T^*B_n\mu^{jv!+s_n}\eta T^*$, for each

$j \geqq 0$. Therefore we can set $t^{(j)} = w_j \begin{bmatrix} p_0, \dots, p_n \\ \alpha_0^{(j)}, \dots, \alpha_n^{(j)} \end{bmatrix}$ for each $j \geqq 0$ so that $t^{(j)}(0) = A\mu^{jv!+v}\eta$, $g(t^{(j)}) \in T^*$ and $t^{(j)} \in T_{jv!+v}$. Since each $\alpha_i^{(j)}$ is embedded in $\alpha_i^{(j+1)}$, we have the condition (b) of the assertion. Thus we get the desired sequence $\{t^{(j)}\}$.

Case II: $K_0 = \phi$

In this case indexes μ^v are not consumed completely, therefore we can set $t^{(j)} = (t/\eta) \circ \mu^{jv!}\eta$ $(j \geqq 0)$ so that $t^{(j)}(0) = A\mu^{jv!+v}\eta$ (where $A = \pi_{1_t}(0))$, $g(t^{(j)}) \in T^*$ and $t^{(j)} \in T_{jv!+v}$. Thus we obtain the desired sequence $\{t^{(j)}\}$ of trees such that $\#(\hat{t}^{(j)}) = \#(\hat{t}^{(j+1)})$ for $j \geqq 0$. Thus we have verified the assertion. We return to the determination of $\{\delta'_j\}$.

Using the assertion, we get a sequence $\{t_i^{(j)}\}$ for each $t_i (1 \leqq i \leqq m)$. By the condition (a) of the assertion we can set $\delta'_j = (\delta \circ \mu^{jv!+v}\eta) \begin{bmatrix} q_1, \dots, q_m \\ t_1^{(i)}, \dots, t_m^{(i)} \end{bmatrix}$ $(j \geqq 0)$ so that $\delta'_j(0) = \delta_{n_j}(0)$, where $n_j = jv! + v$, $g(\delta'_j) \in T^* B \mu^{n_j}\eta T^*$, where $B \in N$ such that $g(\delta_v) \in T^* B \mu^v \eta T^*$. Since each $t_i^{(j)}$ is in T_{n_j}, we have $\#(t_i^{(j)}) \leqq \#(\hat{\theta}) k^{n_j}$ (cf. the proof of Lemma 3.1) and $\#(\hat{\delta}'_j) \leqq \#(\hat{\theta}) k \cdot k^{n_j} = \#(\hat{\theta}) k^{n_j+1}$. We also have $\#(\hat{\delta}'_j) \leqq \#(\hat{\delta}'_{j+1})$ $(j \geqq 0)$ because of the condition (b) of the assertion. Thus the obtained $\{\delta'_j\}$ satisfies the properties (1)$\sim$(3) of Step 1.

Step 2: The construction of $\{\theta_n\}$ $(n \geqq 2)$.

First we notice that we can use δ'_j instead of δ_{n_j} at the n_j-th stage because of the properties (1) and (2) of $\{\delta_j\}$.

Now we determine θ_2. It is necessary to satisfy $\#(\hat{\theta}_1) < \#(\hat{\theta}_2)$. For this purpose, σ_2, χ_2 and ψ_2 are to be determined such that $\#(\hat{\chi}_1) < \#(\sigma_2 \cdot \hat{\chi}_2 \cdot \psi_2)$. Let $n_\theta = [\#(\hat{\chi}_1)/v!] + 2$, then we have $n_\theta \geqq 2$ and $\#(\hat{\chi}_1) < n_\theta v! < n_\theta v! + v$.[10] Considering the $n_\theta v! + v$-th stage, we set $\sigma_2 = \beta_2 \cdot \beta_3 \cdot \dots \cdot \beta_{n_\theta v!+v}$, $\psi_2 = \tau_{n_\theta v!+v} \cdot \dots \cdot \tau_3 \cdot \tau_2$. Since either β_i or τ_i increases at least one terminal or ε (the condition (3) of Lemma 3.1), we have $\#(\hat{\chi}_1) < \#(\hat{\delta}_2) + \#(\psi_2) - 1$. Therefore we set $\chi_2 = \delta'_{n_\theta} = (\delta \circ \mu^{n_\theta v!+v}) \begin{bmatrix} q_1, \dots, q_m \\ t_1^{(n_\theta)}, \dots, t_m^{(n_\theta)} \end{bmatrix}$, and we have $\#(\chi_1) < \#(\sigma_2 \cdot \hat{\chi}_2 \cdot \psi_2)$. Setting $\theta_2 = \rho \cdot \sigma_1 \cdot \sigma_2 \cdot \chi_2 \cdot \psi_2 \cdot \psi_1 \cdot \omega$, we have $\theta_2 \in \mathscr{T}(G)$ and $\#(\hat{\theta}_1) < \#(\hat{\theta}_2)$.

In general, for each $j \geqq 3$ we set $\chi_j = \delta'_{n_\theta+j-2}$, $\sigma_j = \beta_{h+1} \cdot \beta_{h+2} \cdot \dots \cdot \beta_{h+v!}$

10) $[x]$ denotes the greatest integer less than or equal to x.

and $\psi_j = \tau_{h+v!} \cdot \tau_{h+v!-1} \cdot \cdots \cdot \tau_{h+1}$, where $h = (n_\theta + j - 3)v! + v$. Since we can use χ_j instead of δ_m at the m-th stage, where $m = (n_\theta + j - 2)v! + v$, we have $\theta_j = \rho \cdot \sigma_1 \cdot \sigma_2 \cdot \cdots \cdot \sigma_j \cdot \chi_j \cdot \psi_j \cdot \cdots \cdot \psi_2 \cdot \psi_1 \cdot \omega \in \mathscr{T}(G)$ for each $j \geqq 3$. Thus the condition (i) of this lemma is verified. We also have $\#(\hat{\theta}_j) < \#(\hat{\theta}_{j+1})$ $(j \geqq 2)$ because of the condition (3) of Lemma 3.1 and the fact that $\#(\hat{\chi}_j) \leqq \#(\hat{\chi}_{j+1})$ holds (the property (3) of $\{\delta'_j\}$). Thus we obtain the former inequality. Now we clarify the latter inequality. We have $\#(\hat{\chi}_j) = \#(\hat{\delta}'_{n_\theta + j - 2}) \leqq \#(\hat{\theta}) k^{m+1}$ by the property (3) of $\{\delta'_j\}$, where $m = (n_\theta + j - 2)v! + v$. Since ρ and ω are subtrees of θ, we have $\#(\rho)$, $\#(\hat{\omega}) \leqq k_\theta$, where $k_\theta = \#(\hat{\theta})$. The constitution of δ_j and ψ_j implies that we have $\sigma_1 \cdot \sigma_2 \cdot \cdots \cdot \sigma_j = \beta_1 \cdot \beta_2 \cdot \cdots \cdot \beta_m \cdot \psi_j \cdot \cdots \cdot \psi_2 \cdot \psi_1 = \tau_m \cdot \tau_{m-1} \cdot \cdots \cdot \tau_1$. Therefore, using the condition (2) of Lemma 3.1, we obtain

$$
\begin{aligned}
\#(\hat{\theta}_j) &< \#(\hat{\rho}) + \sum_{i=1}^{j} \#(\hat{\rho}_i) + \#(\hat{\chi}_j) + \sum_{i=1}^{j} \#(\psi_j) + \#(\hat{\omega}) \\
&< k_\theta + \sum_{i=1}^{m} \#(\hat{\beta}_i) + k_\theta k^{m+1} + \sum_{i=1}^{m} \#(\hat{\tau}_i) + k_\theta \\
&\leqq 2k_\theta + 2 \sum_{i=1}^{m} k_\theta k^i + k_\theta k^{m+1} \\
&= k_\theta \left(2 + 2 \frac{k^{m+1} - 1}{k - 1} + k^{m+1} \right) \leqq 3 k_\theta k^{m+1} \\
&= 3 \#(\hat{\theta}) k^{(n_\theta + j - 2)v! + v + 1} .
\end{aligned}
$$

Now set the constant $c_\theta = 3\#(\hat{\theta}) k^{(n_\theta - 2)v! + v + 2} > \#(\hat{\theta})$ (because $n_\theta \geqq 2$) and $c = k^{v!} > 1$ (because $k > 1$), then we have $\#(\hat{\theta}_j) < c_\theta c^j$. Thus the condition (11) is satisfied.

5. Main Theorem

In this section, combining Lemma 4.1 with Lemma 2.1, we obtain our main theorem (Theorem 5.1). Next we give its applications.

Theorem 5.1. *Given an indexed grammar $G = (N, T, F, P, S)$, there exist integers k and c with the following property. For any $\theta \in \mathscr{T}(G)$ such that $\#(\hat{\theta}) > k$, there exist trees ρ and ω, for each $i \geqq 1$, σ_i, χ_i and $\psi_i \in \mathscr{D}(G)$, and θ is decomposed into $\rho \cdot \sigma_1 \cdot \chi_1 \cdot \psi_1 \cdot \omega$ and the following conditions* (i) *and*

(ii) *are satisfied*:

(i) *For each* $n \geqq 1$,

$$\theta_n = \rho \cdot \sigma_1 \cdot \sigma_2 \cdot \cdots \cdot \sigma_n \cdot \chi_n \cdot \psi_n \cdot \cdots \cdot \psi_2 \cdot \psi_1 \cdot \omega \in \mathscr{T}(G).$$

(ii) $\#(\hat{\theta}_n) < \#(\hat{\theta}_{n+1}) < c_\theta c^{n+1}$ *(for each* $n \geqq 1$*)*,

where c_θ *is a constant depending on* θ.

Proof. The constants k, c and c_θ are as in Lemma 4.1. If θ is a non-CF-like tree, we apply Lemma 4.1. If θ has a CF-like pair of nodes, we apply Lemma 2.1 to θ. We set $\rho = \alpha$, for each $i \geqq 1$, $\sigma_i = \beta$, $\chi_i = \delta$ and $\psi_i = \Lambda$, and $\omega = \Lambda$, where Λ is the empty tree and α, β and δ are the trees obtained by applying Lemma 2.1. Since the constant c_θ is larger than $\#(\hat{\theta})$ and $c > 1$, the conditions (i) and (ii) of this theorem are satisfied when we use the coresponding conditions (i) and (ii) of Lemma 2.1.

As an immediate corollary of Theorem 5.1, we obtain the solvability of the finiteness problem about indexed languages. This fact has already been proved by Rounds [5]. But here we give another proof following our formulation.

Corollary 5.1. *Given any indexed grammar* G, *the question of whether* $L(G)$ *is finite or not is solvable.*

Proof. Given an indexed grammar $G = (N, T, F, P, S)$, an ε-free[11] indexed grammar $G' = (N', T, F', P', S')$ such that $L(G') = L(G) - \{\varepsilon\}$, can be effectively constructed from G (Aho [1], p661). Clearly $L(G)$ is finite if and only if $L(G')$ is finite. Calculate the constant k of Theorem 5.1 for G'. We now show that if there is a string $w \in L(G')$ with $|w| > k$, there are infinitely many strings in $L(G')$. To see this we apply Theorem 5.1 to $\theta \in \mathscr{T}(G')$ such that $g(\theta) = w$. Since G' is ε-free, we have $\#(\hat{\gamma}) = |g(\gamma)|$ for any $\gamma \in \mathscr{T}(G')$. Therefore the condition (ii) of Theorem 5.1 guarantees $|g(\theta_n)| < |g(\theta_{n+1})|$ for $n \geqq 1$.

To decide whether there exist strings $w \in L(G')$ with $|w| > k$, we let

---

11) An indexed grammar $G = (N, T, F, P, S)$ is called *ε-free*, if $A \to \varepsilon \notin P \cup \bigcup_{f \in F} f$.

$R(k)$ be the regular set of strings $x \in T^*$ such that $|x|>k$, and we consider whether $R(k) \cap L(G')$ is empty. This question is decidable, because an indexed grammar $G''=(N'', T, P'', F'', S'')$ such that $L(G'')=R(k) \cap L(G')$ can be effectively constructed (Aho [1], p.656). The emptiness problem for indexed grammars is solvable (Aho [1], p. 658).

Next we establish a result which states that certain languages are *not* indexed languages.

Theorem5.2. *Let f be a function from J into J such that $f(n) \leqq f(n+1)(n \geqq 1)$ and $\overline{\lim\limits_{n\to\infty}} f(n)^{1/n}=\infty$, then $L_f=\{a^{f(n)} \mid n \geqq 1\}$ is not an indexed language.*

Proof. Without loss of generality, we suppose there is an ε-free indexed grammar G such that $L_f=L(G)$. Let k and c be integers which satisfy Theorem 5.1. Since $\overline{\lim\limits_{n\to\infty}} f(n)^{1/n}=\infty$, there exists an integer n_0 such that $f(n_0)>k$. For the one of such n_0, we may select $\theta \in \mathscr{T}(G)$ satisfying $a^{f(n_0)}=g(\theta)$. Since G is ε-free, $\#(\hat{\theta})=|g(\theta)|=f(n_0)>k$. Therefore we apply Theorem 5.1 to this θ, and calculate an integer c_θ. There exists an integer $t \geqq 1$ such that $f(n_0+t)^{1/(n_0+t)}>c_\theta c^2$, because we have $\overline{\lim\limits_{n\to\infty}} f(n)^{1/n}=\infty$. Using the condition (ii) of Theorem 5.1, we have the following inequalities:

$$f(n_0+t)>(c_\theta c^2)^{n_0+t}>c_\theta c^{2t+1}>\#(\hat{\theta}_{2t+1})=|g(\theta_{2t+1})|>$$

$$\#(\hat{\theta}_{2t})=|g(\theta_{2t})|>\cdots>|g(\theta_1)|=|g(\theta)|=f(n_0).$$

Since f is a monotone function, there are at most $t+1$ distinct elements in L_f whose lengths are between $f(n_0)$ and $f(n_0+t)$. On the other hand, the above inequalities tell us that at least $2t+1$ such elements exist. This is a contradiction. Thus $L(G) \neq L_f$.

Theorem 5.2 tells us that such languages as $\{a^{n!} \mid n \geqq 1\}$ and $\{a^{n^n} \mid n \geqq 1\}$ aren't indexed languages, because $\overline{\lim\limits_{n\to\infty}} (n!)^{1/n}=\overline{\lim\limits_{n\to\infty}} (n^n)^{1/n}=\infty$. On the other hand, if $p_i \in J[X]$ is a polynomial and if $k_i \in J$, then the language $\{a^{\Phi(n)} \mid n \geqq 1\}$ where $\Phi(x)=\sum\limits_{i=1}^{j} p_i(x)k_i^x$, is an indexed language. Taking this point into consideration, Theorem 5.2 may be thought to give a good limit of the indexed grammars about their generative powers.

Using the proof techniques developed for Theorem 5.1, we get the following theorem.

Theorem 5.3. *For an alphabet* Σ, *a language* $L_\Sigma=\{(\$w)^{|w|}\mid w\in\Sigma^*\}$ *is not an indexed language, where* \$ *is a special symbol not in* Σ.

Proof. It is sufficient to show that the existence of an ε-free indexed grammar $G=(N, \Sigma\cup\{\$\}, P, F, S)$ such that $L_\Sigma-\{\varepsilon\}=L(G)$ leads to a contradiction. For this purpose we use the similar construction method developed for Theorem 5.1. It is tedious to repeat the similar definitions and lemmas, we state the necessary alterations.

First the definition of P-node (Definition 2.2) is changed as follows. For $\gamma\in\mathscr{D}(G)$, a node $p\in D_\gamma$ is called a P-node when there exist at least two distinct integers i_1 and i_2 such that each $\gamma/p\cdot i_j$ contains at least one node whose label is \$. We can show that there exists a constant k' depending on G with the following property. For any non-CF-like tree $\gamma\in\mathscr{T}(G)$ such that $\#_\$(\hat{\gamma})>k'$, ($\#_\$(\hat{\gamma})$ is the number of nodes in $\hat{\gamma}$ whose labels are \$.), there exists a chain C of γ in which there exist seven B-nodes $p_{-5}<p_{-4}\cdots<p_{-1}<p_0<p_1$ whose h-function values are the same. The constant k' is equal to m^{k_1-1}, where $k_1=vk_2$, $k_2=\dfrac{(s+1)k_3'^{+1}-1}{s}$ and $k_3'=6v(v+1)(t+1)2^v+1$. The constants m, v, s and t depending on G are as follows: $m=r(G)$; $v=\#(N)$; $s=\max\{|\eta_i|\mid A\to X_1\eta_1X_2\eta_2\cdots X_k\eta_k\in P\}$; $t=\#(F)$. To verify the above claim, we have only to repeat the same construction method up to Lemma 2.2. Namely for $\gamma\in\mathscr{T}(G)$ such that $\#_\$(\hat{\gamma})>k'$, let C be one of the chains which contain the maximum number of P-nodes, then there exist at least k_1 P-nodes in C. Let M be one of the P-mountains which have most the P-nodes, then there are at least k_2 P-nodes in M. We form a tree γ_M whose labels consist of elements in M. Let $\bar{C}$ be one of the chains of γ_M which contain the maximum number of B-nodes, then there exist at least k_3' B-nodes in $\bar{C}$. Since the h-function has at most $v(v+1)(t+1)2^v$ distinct values, there exist seven B-nodes $p_{-5}<p_{-4}\cdots<p_{-1}<p_0<p_1$ whose h-function values are the same.

Using only p_0 and p_1, we get the decomposition of $\gamma=\alpha\cdot\beta_1\cdot\delta_1\cdot\tau_1\cdot\nu$. The existence of the other five B-nodes $p_{-5},\dots, p_{-1}$ implies that either α or ν has at least three P-nodes. When we apply the algorithms described at Lemma 3.1 and Lemma 4.1 to γ, we get a sequence of trees $\{\theta_n\}$ in $\mathscr{T}(G)$ such that $\sharp(\hat{\theta}_n)<\sharp(\hat{\theta}_{n+1})$ (for $n\geqq 1$), $\theta_1=\gamma$ and each θ_n has α and ν as its components.

Given an arbitrary integer i, there are infinitely many elements in L_Σ which have more than i occurrences of \$. we have $\gamma\in\mathscr{T}(G)$ such that $\sharp_\$(\hat{\gamma})>k'$. We fix such a γ. If γ is a non-CF-like tree, we can get the above sequence of trees $\{\theta_n\}$. Then we have a sequence $\{g(\theta_n)\}$ of elements in L_Σ such that $|g(\theta_n)|<|g(\theta_{n+1})|(n\geqq 1)$. Let $g(\gamma)=g(\theta_1)=(\$w_0)^{|w_0|}$, then there exists a substring $\cdots\$w_0\$\cdots$ in $g(\alpha)$ or $g(\nu)$ because either α or ν has at least three P-nodes. This leads to a contradiction since each θ_n has α and ν as its components.

If γ has a CF-like pair of nodes, apply Lemma 2.1 to γ. Since β_1 is repeated in this case and β_1 has at least one P-node, we have an element in L_Σ which has a substring $\cdots(w_1\$w_2)^n\cdots$ for each $n\geqq 1$. This leads to a contradiction again. Therefore $L_\Sigma-\{\varepsilon\}\neq L(G)$.

This theorem is also interesting, since the languages $L_m=\{(\$w)^m\,|\,w\in\Sigma^*\}$ and $L=\bigcup_{m=0}^{\infty} L_m$ are indexed languages.

Acknowledgement

The author expresses his gratitude to Professor Satoru Takasu for his criticism and encouragement. The author also wishes to thank Mr. Teruyasu Nishizawa of the University of Electro-Communications and Mr. Takumi Kasai for their helpful suggestions and discussions.

References

[1] Aho, A. V., Indexed grammars, *J. Assoc. Comput. Mach.* **15** (1968), 647–671.

[2] Bar-Hillel et al., On formal properties of simple prase structure grammars, *Z. Phonetik Sprachwiss. Kommunikat.* **14** (1961), 143–172.

[3] Brainerd, W. S., Tree generating regular systems, *Infomation and Control* **14** (1969), 217-231.

[4] Ogden, W. F., Intercalation theorems for pushdown store and stack languages, Thesis, Stanford Univ. (1968), also in *Proc. 1st ACM Symp. Theory on Computing* (1969), 31-42,

[5] Rounds, W. F., Tree oriented proofs of some theorems in context-free and indexed languages, *Proc. 2nd ACM Symp. Theory on Computing* (1970), 109-116.
[6] Takahashi, M., Derivation trees and context-free languages, *Masters' thesis, Pennsylvania Univ.* (1970).