PAPER

Analogical Conception of Chomsky Normal Form and Greibach Normal Form for Linear, Monadic Context-Free Tree Grammars

Akio FUJIYOSHI[†a)], ***Member***

SUMMARY This paper presents the analogical conception of Chomsky normal form and Greibach normal form for linear, monadic context-free tree grammars (LM-CFTGs). LM-CFTGs generate the same class of languages as four well-known mildly context-sensitive grammars. It will be shown that any LM-CFTG can be transformed into equivalent ones in both normal forms. As Chomsky normal form and Greibach normal form for context-free grammars (CFGs) play a very important role in the study of formal properties of CFGs, it is expected that the Chomsky-like normal form and the Greibach-like normal form for LM-CFTGs will provide deeper analyses of the class of languages generated by mildly context-sensitive grammars.

key words: *formal languages, tree adjoining grammar, mildly context-sensitive grammar, context-free tree grammar*

1. Introduction

Recently, the class of grammar formalisms called mildly context-sensitive grammars has been investigated very actively. Since it was shown that tree adjoining grammars [1], [6], [7], combinatory categorial grammars, linear indexed grammars, and head grammars generate the same class of languages [12], the class of languages generated by these mildly context-sensitive grammars has been thought to be very important in the theory of formal languages. The languages $\{a^n b^n c^n d^n \mid n \geq 0\}$ and $\{ww \mid w \in \{a, b\}^*\}$ can be generated by these formalisms, whereas neither of them can be generated by a context-free grammar (CFG). It is noteworthy that the class of languages generated by these formalisms can be recognized in $O(n^6)$ or $O(M(n^2))$ time [9], [10].

A restricted version of context-free tree grammars (CFTGs) [11] which generates the same class of languages as the above four grammar formalisms is linear, monadic context-free tree grammars (LM-CFTGs) [2], [5]. An LM-CFTG is a CFTG where the number of occurrences of every variable in the right-hand side of a production is no more than 1 and the ranks of nonterminals are either 0 or 1.

This paper focuses on LM-CFTGs and presents analogical conception of Chomsky normal form and Greibach normal form for LM-CFTGs. It will be shown that any LM-CFTG can be transformed into equivalent ones in both normal forms. The form of productions of a grammar in each normal form is considerably simple. As Chomsky normal form and Greibach normal form for CFGs play a very important role in the study of formal properties of CFGs, it is expected that the Chomsky-like normal form and the Greibach-like normal form for LM-CFTGs will provide deeper analyses of the class of languages generated by mildly context-sensitive grammars.

Manuscript received February 17, 2006.
Manuscript revised July 5, 2006.
[†]The author is with the Department of Computer and Information Sciences, Ibaraki University, Hitachi-shi, 316–8511 Japan.
a) E-mail: fujiyosi@mx.ibaraki.ac.jp
DOI: 10.1093/ietisy/e89–d.12.2933

2. Preliminaries

In this section, some terms, definitions and former results which will be used in the rest of this paper are introduced.

2.1 Ranked Alphabets, Trees and Substitution

A *ranked alphabet* is a finite set of symbols in which each symbol is associated with a natural number, called the *rank* of a symbol. Let Σ be a ranked alphabet. For $n \geq 0$, let $\Sigma_n = \{a \in \Sigma \mid \text{the rank of } a \text{ is } n\}$.

The set T_Σ (*trees* over Σ) is the smallest set of strings over Σ, parentheses and commas such that (1) $\Sigma_0 \subseteq T_\Sigma$ and (2) if $\alpha_1, \alpha_2, \ldots, \alpha_n \in T_\Sigma$ and $a \in \Sigma_n$ for some $n \geq 1$, then $a(\alpha_1, \alpha_2, \ldots, \alpha_n) \in T_\Sigma$.

Let λ be the empty string. Let ε be the special symbol that may be contained in Σ_0. The *yield* of a tree is a function from T_Σ into Σ^* defined as follows. For $\alpha \in T_\Sigma$, (1) if $\alpha = a \in (\Sigma_0 - \{\varepsilon\})$, then $\text{yield}(\alpha) = a$, (1') if $\alpha = \varepsilon$, then $\text{yield}(\alpha) = \lambda$, and (2) if $\alpha = a(\alpha_1, \alpha_2, \ldots, \alpha_n)$ for some $a \in \Sigma_n$ and $\alpha_1, \alpha_2, \ldots, \alpha_n \in T_\Sigma$, then $\text{yield}(\alpha) = \text{yield}(\alpha_1) \cdot \text{yield}(\alpha_2) \cdot \cdots \cdot \text{yield}(\alpha_n)$.

Let $X = \{x_1, x_2, \ldots\}$ be the fixed countable set of variables. Let $X_0 = \emptyset$ and for $n \geq 1$, let $X_n = \{x_1, x_2, \ldots, x_n\}$. x_1 is situationally denoted by x. $T_\Sigma(X_n)$ is defined to be $T_{\Sigma \cup X_n}$ taking the ranks of elements in X are all 0. For $\alpha \in T_\Sigma(X_n)$ and $\beta_1, \beta_2, \ldots, \beta_n \in T_\Sigma(X)$, $\alpha[\beta_1, \beta_2, \ldots, \beta_n]$ is defined to be the result of substituting each β_i $(1 \leq i \leq n)$ for the occurrences of the variable x_i in α.

A tree $\alpha \in T_\Sigma(X_n)$ is *linear* if no variable occurs more than once in α. A tree $\alpha \in T_\Sigma(X_n)$ is *nondeleting* if all variables in X_n occur at least once in α. The set of all linear trees and all nondeleting trees in $T_\Sigma(X_n)$ are denoted by $T_\Sigma\lceil X_n \rceil$ and $T_\Sigma\lfloor X_n \rfloor$, respectively.

2.2 Context-Free Tree Grammars

The context-free tree grammars (CFTGs) were introduced by W. C. Rounds [11] as tree generating systems. The definition of CFTGs is a direct generalization of context-free

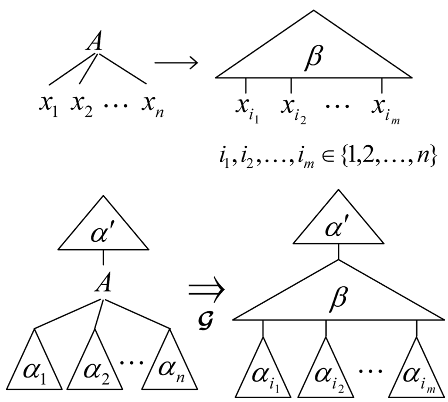

Fig. 1 One-step derivation.

grammars (CFGs).

A *context-free tree grammar* (CFTG) is a four-tuple $\mathcal{G} = (N, \Sigma, P, S)$, where:

- N and Σ are disjoint ranked alphabets of *nonterminals* and *terminals*, respectively.
- P is a finite set of *productions* of the form $A(x_1, x_2, \ldots, x_n) \to \alpha$ with $n \geq 0$, $A \in N_n$ and $\alpha \in T_{N\cup\Sigma}(X_n)$. For $A \in N_0$, productions are written as $A \to \alpha$ instead of $A() \to \alpha$.
- S, the *initial nonterminal*, is a distinguished symbol in N_0.

For a CFTG $\mathcal{G}$, the *one-step derivation* $\underset{\mathcal{G}}{\Rightarrow}$ is the relation on $T_{N\cup\Sigma}(X) \times T_{N\cup\Sigma}(X)$ such that for a tree $\alpha \in T_{N\cup\Sigma}(X)$, if $\alpha = \alpha'[A(\alpha_1, \alpha_2, \ldots, \alpha_n)]$ for some $\alpha' \in T_{N\cup\Sigma}\lceil X_1 \rceil \cap T_{N\cup\Sigma}\lfloor X_1 \rfloor$, $A \in N_n$ and $\alpha_1, \alpha_2, \ldots, \alpha_n \in T_{N\cup\Sigma}(X)$, and $A(x_1, x_2, \ldots, x_n) \to \beta$ is in P, then $\alpha \underset{\mathcal{G}}{\Rightarrow} \alpha'[\beta[\alpha_1, \alpha_2, \ldots, \alpha_n]]$. See Fig. 1.

An *(n-step) derivation* is a finite sequence of trees $\alpha_0, \alpha_1, \ldots, \alpha_n \in T_{N\cup\Sigma}(X)$ such that $n \geq 0$ and $\alpha_0 \underset{\mathcal{G}}{\Rightarrow} \alpha_1 \underset{\mathcal{G}}{\Rightarrow} \cdots \underset{\mathcal{G}}{\Rightarrow} \alpha_n$. When there exists a derivation $\alpha_0, \alpha_1, \ldots, \alpha_n$, we write $\alpha_0 \overset{n}{\underset{\mathcal{G}}{\Rightarrow}} \alpha_n$ or $\alpha_0 \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \alpha_n$.

The *tree language generated by* $\mathcal{G}$ is the set $L(\mathcal{G}) = \left\{\alpha \in T_\Sigma \,\middle|\, S \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \alpha\right\}$. The *language generated by* $\mathcal{G}$ is $L_S(\mathcal{G}) = \{\text{yield}(\alpha) \,|\, \alpha \in L(\mathcal{G})\}$. Note that $L_S(\mathcal{G}) \subseteq (\Sigma_0 - \{\varepsilon\})^*$.

Let $\mathcal{G}$ and $\mathcal{G}'$ be CFTGs. $\mathcal{G}$ and $\mathcal{G}'$ are *equivalent* if $L(\mathcal{G}) = L(\mathcal{G}')$. $\mathcal{G}$ and $\mathcal{G}'$ are *weakly equivalent* if $L_S(\mathcal{G}) = L_S(\mathcal{G}')$.

2.3 Restrictions on Context-Free Tree Grammars

We introduce restrictions on CFTGs and former results about subclasses of CFTGs.

A CFTG $\mathcal{G} = (N, \Sigma, P, S)$ is *monadic* if the rank of any nonterminal is 0 or 1, i.e., $N = N_0 \cup N_1$ and $N_n = \emptyset$ for $n \geq 2$. $\mathcal{G}$ is *linear* if for any production $A(x_1, x_2, \ldots, x_n) \to \alpha$ in P, $\alpha \in T_{N\cup\Sigma}\lceil X_n \rceil$. $\mathcal{G}$ is *nondeleting* if for any production $A(x_1, x_2, \ldots, x_n) \to \alpha$ in P, $\alpha \in T_{N\cup\Sigma}\lfloor X_n \rfloor$.

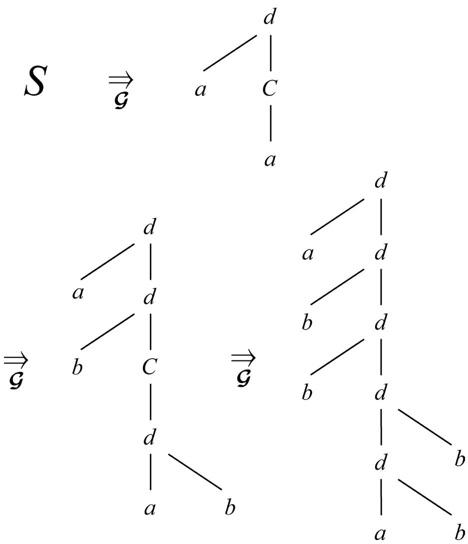

Fig. 2 A derivation of a tree in $L(\mathcal{G})$.

When $\mathcal{G}$ is monadic, all productions are either of the form $A(x) \to \alpha$ with $A \in N_1$ and $\alpha \in T_{N\cup\Sigma}(X_1)$ or of the form $B \to \beta$ with $B \in N_0$ and $\beta \in T_{N\cup\Sigma}$. When $\mathcal{G}$ is monadic and linear, for any production $A(x) \to \alpha$ with $A \in N_1$, there exists at most one occurrence of x in α. When $\mathcal{G}$ is monadic, linear and nondeleting, for any production $A(x) \to \alpha$ with $A \in N_1$, there exists exactly one occurrence of x in α.

Among subclasses of CFTGs, the following results are known.

Theorem 2.1: (Fujiyoshi [2]) The class of tree languages generated by linear, monadic CFTGs is the same as that generated by linear, nondeleting, monadic CFTGs.

Theorem 2.2: (Fujiyoshi [2]) The class of tree languages generated by monadic CFTGs is properly larger than that generated by linear, monadic CFTGs.

Linear, monadic CFTGs (LM-CFTGs) are related to tree adjoining grammars [1], [6], [7], one of the most famous and well-studied mildly context-sensitive grammar formalisms.

Theorem 2.3: (Fujiyoshi & Kasai [5]) The class of languages generated by LM-CFTGs coincides with that generated by tree adjoining grammars.

Example 2.4: The following $\mathcal{G}$ is an LM-CFTG that generates the language $L_{ww} = \{ww \,|\, w \in \{a, b\}^+\}$. $\mathcal{G} = (N, \Sigma, P, S)$, where $N_0 = \{S\}$, $N_1 = \{C\}$, $\Sigma = \Sigma_0 \cup \Sigma_2$, $\Sigma_0 = \{a, b\}$, $\Sigma_2 = \{d\}$, and P consists of the following productions:

$$
\begin{aligned}
&S \to d(a, a), \quad C(x) \to d(a, d(x, a)),\\
&S \to d(b, b), \quad C(x) \to d(b, d(x, b)),\\
&S \to d(a, C(a)), \quad C(x) \to d(a, C(d(x, a))),\\
&S \to d(b, C(b)), \text{ and } C(x) \to d(b, C(d(x, b))).
\end{aligned}
$$

In Fig. 2, a derivation of a tree in $L(\mathcal{G})$ is illustrated. The yield of the tree is "abbabb."

3. Chomsky-Like Normal Form for LM-CFTGs

In this section, we define Chomsky-like normal form for LM-CFTGs and show that any LM-CFTG can be transformed into an equivalent one in Chomsky-like normal form.

Definition 3.1: An LM-CFTG $\mathcal{G} = (N, \Sigma, P, S)$ is in *Chomsky-like normal form* if P consists of productions in one of the following forms:

(1) $A \to B(C)$ with $A, C \in N_0$ and $B \in N_1$,
(2) $A \to a$ with $A \in N_0$ and $a \in \Sigma_0$,
(3) $A(x) \to B(C(x))$ with $A, B, C \in N_1$, or
(4) $A(x) \to b(C_1, \ldots, C_{i-1}, x, C_{i+1}, \ldots, C_n)$ with $A \in N_1$, $n \geq 1$, $b \in \Sigma_n$, $1 \leq i \leq n$ and $C_1, \ldots, C_{i-1}, C_{i+1}, \ldots, C_n \in N_0$.

See Fig. 3.

Intuitively, a production of an LM-CFTG has two functions of growing trees: growing trees horizontally and growing trees vertically. The vertical growth of a tree is like a derivation of a string by CFGs, but on the other hand, the horizontal growth of a tree is peculiar to tree grammars. If we ignore the function of growing trees horizontally, the productions (1) and (3) correspond to the context-free production $A \to BC$, and (2) and (4) correspond to $A \to a$ and $A \to b$, respectively. That is why we call it "Chomsky-like normal form."

Example 3.2: The following $\mathcal{G}'$ is an LM-CFTG in Chomsky-like normal form that is equivalent to $\mathcal{G}$ in Example 2.4. $\mathcal{G}' = (N', \Sigma, P', S)$, where $N'_0 = \{S, A, B\}$, $N'_1 = \{C, D_1, D_2, D_3, D_4, E_1, E_2\}$, $\Sigma = \Sigma_0 \cup \Sigma_2$, $\Sigma_0 = \{a, b\}$, $\Sigma_2 = \{d\}$, and P' consists of the following productions:

$$S \to D_1(A), \quad S \to E_1(A), \quad A \to a,$$
$$S \to D_2(B), \quad S \to E_2(B), \quad B \to b,$$
$$E_1(x) \to D_1(C(x)), \quad E_2(x) \to D_2(C(x)),$$
$$C(x) \to D_1(D_3(x)), \quad C(x) \to D_2(D_4(x)),$$
$$C(x) \to E_1(D_3(x)), \quad C(x) \to E_2(D_4(x)),$$
$$D_1(x) \to d(A, x), \quad D_2(x) \to d(B, x),$$
$$D_3(x) \to d(x, A), \text{ and } D_4(x) \to d(x, B).$$

In Fig. 4, a derivation of a tree in $L(\mathcal{G}')$ is illustrated.

Theorem 3.3: For any LM-CFTG $\mathcal{G} = (N, \Sigma, P, S)$, we can construct an equivalent LM-CFTG in Chomsky-like normal form.

Proof. Without loss of generality, we may assume that $\mathcal{G}$ is in normal form presented in [5], so P consists of productions in one of the following forms:

(i) $A \to B(C)$ with $A, C \in N_0$ and $B \in N_1$,
(ii) $A \to a$ with $A \in N_0$ and $a \in \Sigma_0$,
(iii) $A(x) \to B_1(B_2(\cdots(B_m(x))\cdots))$ with $A \in N_1$, $m \geq 0$ and $B_1, B_2, \ldots, B_m \in N_1$, or

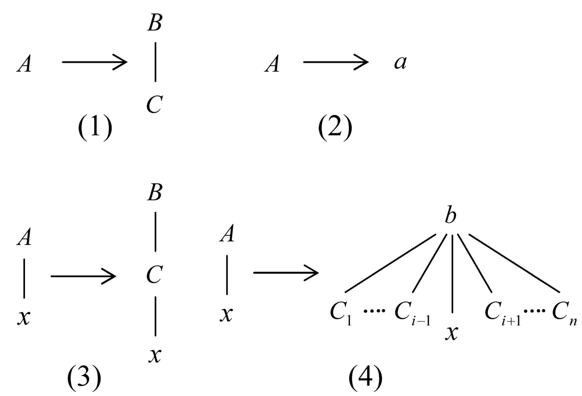

Fig. 3 Chomsky-like normal form.

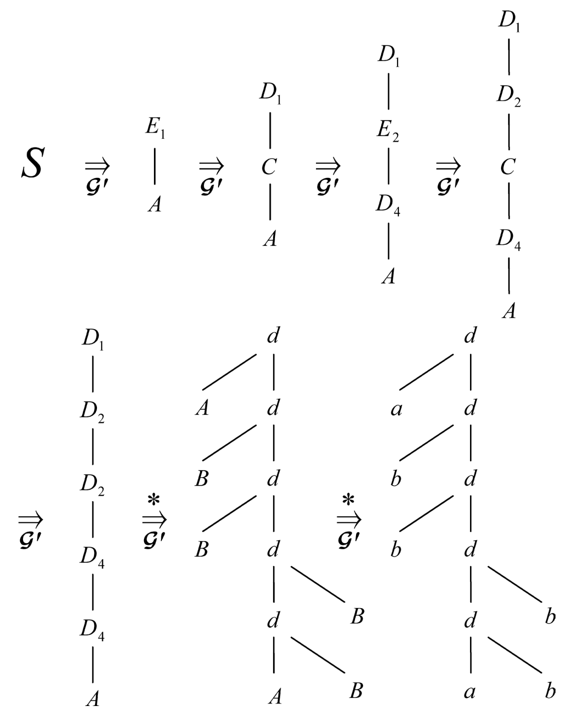

Fig. 4 A derivation of a tree in $L(\mathcal{G}')$.

(iv) $A(x) \to b(C_1, \ldots, C_{i-1}, x, C_{i+1}, \ldots, C_n)$ with $A \in N_1$, $n \geq 1$, $b \in \Sigma_n$, $1 \leq i \leq n$ and $C_1, \ldots, C_{i-1}, C_{i+1}, \ldots, C_n \in N_0$.

First, we replace productions of the form (iii) above with $m \geq 3$. Let $A(x) \to B_1(B_2(\cdots(B_m(x))\cdots))$ be a production with $m \geq 3$. Then introduce a new nonterminal B' of rank 1, and replace the production by the two productions $A(x) \to B_1(B'(x))$ and $B'(x) \to B_2(B_3(\cdots(B_m(x))\cdots))$. Repeat this operation as many times as possible. Let N' and P' be the resulting nonterminals and productions, respectively. The LM-CFTG $\mathcal{G}' = (N', \Sigma, P', S)$ is clearly equivalent to $\mathcal{G}$.

Second, we replace productions of the form $A(x) \to x$ with $A \in \Sigma_1$. Let $\hat{P}$ be the smallest set satisfying the following conditions:

- $P' \subseteq \hat{P}$.
- If $A(x) \to B(C(x))$ is in $\hat{P}$ and $B(x) \to x$ is in $\hat{P}$, then $A(x) \to C(x)$ is in $\hat{P}$.
- If $A(x) \to B(C(x))$ is in $\hat{P}$ and $C(x) \to x$ is in $\hat{P}$, then $A(x) \to B(x)$ is in $\hat{P}$.
- If $A(x) \to B(x)$ is in $\hat{P}$ and $B(x) \to x$ is in $\hat{P}$, then $A(x) \to x$ is in $\hat{P}$.
- If $A \to B(C)$ is in $\hat{P}$ and $B(x) \to x$ is in $\hat{P}$, then $A \to C$ is in $\hat{P}$.

Let P'' be the set obtained from $\hat{P}$ removing all productions of the form $A(x) \to x$. Then P'' consists of productions in one of the following forms:

(a) $A \to B$ with $A, B \in N_0$,
(b) $A \to B(C)$ with $A, C \in N_0$ and $B \in N_1$,
(c) $A \to a$ with $A \in N_0$ and $a \in \Sigma_0$,
(d) $A(x) \to B(x)$ with $A, B \in N_1$,
(e) $A(x) \to B(C(x))$ with $A, B, C \in N_1$, or
(f) $A(x) \to b(C_1, \ldots, C_{i-1}, x, C_{i+1}, \ldots, C_n)$ with $A \in N_1$, $n \geq 1$, $b \in \Sigma_n$, $1 \leq i \leq n$ and $C_1, \ldots, C_{i-1}, C_{i+1}, \ldots, C_n \in N_0$.

The LM-CFTG $\mathcal{G}'' = (N', \Sigma, P'', S)$ is clearly equivalent to $\mathcal{G}'$.

Third, we replace productions of the form (a) and (d) above. Let $\tilde{P}$ be the smallest set satisfying the following conditions:

- $P'' \subseteq \tilde{P}$.
- If $A(x) \underset{\mathcal{G}''}{\Rightarrow} B(x)$ for some $A, B \in N_1$ and $A(x) \to \alpha$ is in P'', then $B(x) \to \alpha$ is in $\tilde{P}$.
- If $A \underset{\mathcal{G}''}{\Rightarrow} B$ for some $A, B \in N_0$ and $A \to \alpha$ is in P'', then $B \to \alpha$ is in $\tilde{P}$.

Let P''' be the set obtained from $\tilde{P}$ removing all productions of the form $A \to B$ and all productions of the form $A(x) \to B(x)$. The LM-CFTG $\mathcal{G}''' = (N', \Sigma, P''', S)$ is clearly equivalent to $\mathcal{G}''$. It is clear that $\mathcal{G}'''$ is in Chomsky-like normal form and equivalent to $\mathcal{G}$. □

4. Greibach-Like Normal Form for LM-CFTGs

In this section, we define Greibach-like normal form for LM-CFTGs and show that any LM-CFTG can be transformed into an equivalent one in Greibach-like normal form. In the construction of an equivalent LM-CFTG in Greibach-like normal form, the famous technique to construct a context-free grammar (CFG) in Greibach normal form [8] is employed. The technique can be adapted to LM-CFTGs because paths of derivation trees of LM-CFTGs can be similarly treated as derivation strings of CFGs.

Definition 4.1: An LM-CFTG $\mathcal{G} = (N, \Sigma, P, S)$ is in *Greibach-like normal form* if P consists of productions in one of the following forms:

(1) $A \to a$ with $A \in N_0$ and $a \in \Sigma_0$,

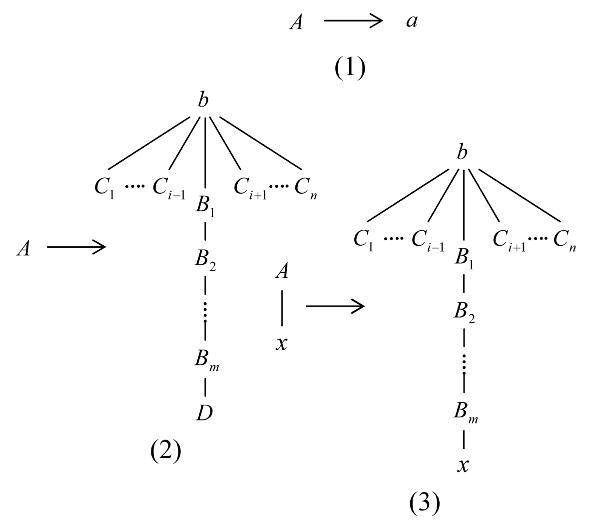

Fig. 5 Greibach-like normal form.

(2) $A \to b(C_1, \ldots, C_{i-1}, \gamma, C_{i+1}, \ldots, C_n)$ with $A \in N_0$, $n \geq 1$, $b \in \Sigma_n$, $1 \leq i \leq n$, $C_1, \ldots, C_{i-1}, C_{i+1}, \ldots, C_n \in N_0$ and $\gamma \in T_N$, or
(3) $A(x) \to b(C_1, \ldots, C_{i-1}, \gamma, C_{i+1}, \ldots, C_n)$ with $A \in N_1$, $n \geq 1$, $b \in \Sigma_n$, $1 \leq i \leq n$, $C_1, \ldots, C_{i-1}, C_{i+1}, \ldots, C_n \in N_0$, and $\gamma \in T_{N_1}(X_1)$.

See Fig. 5. In Fig. 5, $B_1(B_2(\cdots(B_m(D))\cdots))$ is in T_N, and $B_1(B_2(\cdots(B_m(x))\cdots))$ is in $T_{N_1}(X_1)$. Note that m may be 0.

If we ignore the function of growing trees horizontally, the productions (1), (2) and (3) correspond to the context-free productions in Greibach normal form $A \to a$, $A \to bB_1B_2\cdots B_mD$ and $A \to bB_1B_2\cdots B_m$, respectively.

Example 4.2: The following $\mathcal{G}''$ is an LM-CFTG in Greibach-like normal form that is equivalent to $\mathcal{G}$ in Example 2.4. $\mathcal{G}'' = (N'', \Sigma, P'', S)$, where $N_0 = \{S, A, B\}$, $N_1 = \{C, D_3, D_4,\}$, $\Sigma = \Sigma_0 \cup \Sigma_2$, $\Sigma_0 = \{a, b\}$, $\Sigma_2 = \{d\}$, and P' consists of the following productions:

$$S \to d(A, A), \quad S \to d(A, C(A)), \quad A \to a,$$
$$S \to d(B, B), \quad S \to d(B, C(B)), \quad B \to b,$$
$$C(x) \to d(A, D_3(x)), \quad C(x) \to d(A, C(D_3(x))),$$
$$C(x) \to d(B, D_4(x)), \quad C(x) \to d(B, C(D_4(x))),$$
$$D_3(x) \to d(x, A), \text{ and } D_4(x) \to d(x, B).$$

In Fig. 6, a derivation of a tree in $L(\mathcal{G}'')$ is illustrated.

Theorem 4.3: For any LM-CFTG $\mathcal{G} = (N, \Sigma, P, S)$, we can construct an equivalent LM-CFTG in Greibach-like normal form.

Proof. Without loss of generality, we may assume that $\mathcal{G}$ is in Chomsky-like normal form, so P consists of productions in one of the following forms:

(i) $A \to B(C)$ with $A, C \in N_0$ and $B \in N_1$,
(ii) $A \to a$ with $A \in N_0$ and $a \in \Sigma_0$,
(iii) $A(x) \to B(C(x))$ with $A, B, C \in N_1$, or

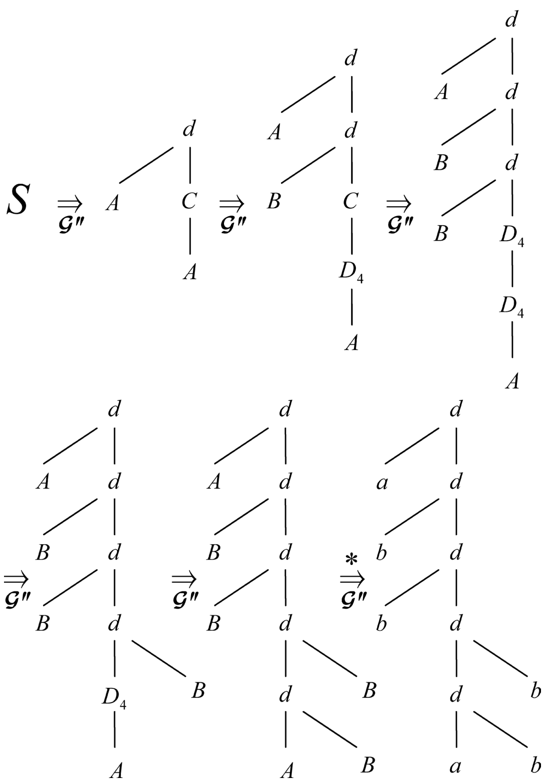

Fig. 6 A derivation of a tree in $L(\mathcal{G}'')$.

(iv) $A(x) \to b(C_1, \ldots, C_{i-1}, x, C_{i+1}, \ldots, C_n)$ with $A \in N_1$, $n \geq 1$, $b \in \Sigma_n$, $1 \leq i \leq n$ and $C_1, \ldots, C_{i-1}, C_{i+1}, \ldots, C_n \in N_0$.

From productions of the form (iii) and (iv) above, we construct a CFG $\mathcal{G}_{\mathrm{cnf}}$ as follows. Let $\mathrm{RHS}_{(\mathrm{iv})}$ be the set of all right-hand side of productions of the form (iv) above. $\mathcal{G}_{\mathrm{cnf}} = (N_1, T, P_{\mathrm{cnf}}, S)$, where N_1 is nonterminals, $T = \{\bar{\alpha} \mid \alpha \in \mathrm{RHS}_{(\mathrm{iv})}\}$ is terminals, $S \in N_1$ is the start symbol, and P_{cnf} is context-free productions defined as follows.

$$\begin{aligned} P_{\mathrm{cnf}} = &\{A \to BC \mid A(x) \to B(C(x)) \in P\} \\ &\cup \{A \to \bar{\alpha} \mid \alpha \in \mathrm{RHS}_{(\mathrm{iv})} \text{ and } A(x) \to \alpha \in P\} \end{aligned}$$

Clearly, the CFG $\mathcal{G}_{\mathrm{cnf}}$ is in Chomsky normal form. Thus we can construct an equivalent CFG $\mathcal{G}_{\mathrm{gnf}} = (N_{\mathrm{gnf}}, T, P_{\mathrm{gnf}}, S)$ which satisfies the following conditions [8]:

- $\mathcal{G}_{\mathrm{gnf}}$ is in Greibach normal form.
- $N_1 \subseteq N_{\mathrm{gnf}}$.
- For any $A \in N_1$ and $w \in T^*$, $A \overset{*}{\underset{\mathcal{G}_{\mathrm{cnf}}}{\Rightarrow}} w$ if and only if $A \overset{*}{\underset{\mathcal{G}_{\mathrm{gnf}}}{\Rightarrow}} w$.

Then, we can construct an LM-CFTG $\mathcal{G}'$ equivalent to $\mathcal{G}$ as follows. $\mathcal{G}' = (N', \Sigma, P', S)$, where $N' = N'_0 \cup N'_1$, $N'_0 = N_0$, $N'_1 = N_{\mathrm{gnf}}$, and P' is the smallest set satisfying the following conditions:

- If $A \to B(C)$ is in P, then $A \to B(C)$ is in P'.
- If $A \to a$ is in P, then $A \to a$ is in P'.
- If $A \to \bar{\alpha} B_1 B_2 \cdots B_m$ is in P_{gnf} with $A \in N_{\mathrm{gnf}}$, $\bar{\alpha} \in T$, $m \geq 0$, and $B_1, B_2, \ldots, B_m \in N_{\mathrm{gnf}}$, then $A(x) \to \alpha[B_1(B_2(\cdots(B_m(x))\cdots))]$ is in P'.

Note that P' consists of productions in one of the following forms:

- $A \to B(C)$ with $A, C \in N_0$ and $B \in N_1$,
- $A \to a$ with $A \in N_0$ and $a \in \Sigma_0$, or
- $A(x) \to b(C_1, \ldots, C_{i-1}, \gamma, C_{i+1}, \ldots, C_n)$ with $A \in N_1$, $n \geq 1$, $b \in \Sigma_n$, $1 \leq i \leq n$, $C_1, \ldots, C_{i-1}, C_{i+1}, \ldots, C_n \in N_0$, and $\gamma \in T_{N_1}(X_1)$.

To show $L(\mathcal{G}') = L(\mathcal{G})$ we prove the following (1) and (2) hold by induction on the length of derivation.

(1) For any $A \in N_0$ and $\alpha \in T_\Sigma$, $A \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \alpha$ if and only if $A \overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \alpha$.

(2) For any $A \in N_1$ and $\alpha \in T_\Sigma(X_1)$, $A(x) \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \alpha$ if and only if $A(x) \overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \alpha$.

The "only if" part is proved as follows. **Basis:** (1) If $A \underset{\mathcal{G}}{\Rightarrow} \alpha$, then $A \to \alpha$ is in P and $\alpha \in \Sigma_0$. Therefore $A \to \alpha$ is in P' and $A \underset{\mathcal{G}'}{\Rightarrow} \alpha$. (2) If $A(x) \underset{\mathcal{G}}{\Rightarrow} \alpha$, then $A(x) \to \alpha$ is in P and $\alpha = b(x)$ for some $b \in \Sigma_1$. By the construction of P', $A(x) \to \alpha$ is in P'. Therefore $A(x) \underset{\mathcal{G}'}{\Rightarrow} \alpha$. **Induction:** For $k \geq 2$, assume that the statement holds for any derivation of length less than k. (1) Suppose that $A \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \alpha$ is a derivation of length k. Then $A \underset{\mathcal{G}}{\Rightarrow} B(C) \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \beta[\gamma] = \alpha$ for some $B \in N_1$, $C \in N_0$, $\beta \in T_\Sigma(X_1)$, and $\gamma \in T_\Sigma$ such that $A(x) \to B(C(x))$ is in P, $B(x) \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \beta$, and $C \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \gamma$. By the induction hypothesis, $B(x) \overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \beta$ and $C \overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \gamma$. By the construction of P', $A \to B(C)$ is in P'. Therefore $A \underset{\mathcal{G}'}{\Rightarrow} B(C) \overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \beta[\gamma] = \alpha$. (2) Suppose that $A(x) \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \alpha$ is a derivation of length k. Then the following derivation is possible for some $m \geq 1$, $B_1, B_2, \ldots, B_m \in N_1$, $\beta_1, \beta_2, \ldots, \beta_m \in \mathrm{RHS}_{(\mathrm{iv})}$, and $\alpha_1, \alpha_2, \ldots, \alpha_m \in T_\Sigma(X_1)$ such that for $1 \leq i \leq m$, $B_i(x) \to \beta_i$ is in P and $B_i(x) \underset{\mathcal{G}}{\Rightarrow} \beta_i \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \alpha_i$.

$$\begin{aligned} A(x) &\overset{*}{\underset{\mathcal{G}}{\Rightarrow}} B_1(B_2(\cdots(B_m(x))\cdots)) \\ &\overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \beta_1[\beta_2[\cdots[\beta_m]\cdots]] \\ &\overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \alpha_1[\alpha_2[\cdots[\alpha_m]\cdots]] = \alpha \end{aligned}$$

By the induction hypothesis, for $1 \leq i \leq m$, $\beta_i \overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \alpha_i$. Because $A \overset{*}{\underset{\mathcal{G}_{\mathrm{cnf}}}{\Rightarrow}} \bar{\beta}_1\bar{\beta}_2\cdots\bar{\beta}_m$, $A \overset{*}{\underset{\mathcal{G}_{\mathrm{gnf}}}{\Rightarrow}} \bar{\beta}_1\bar{\beta}_2\cdots\bar{\beta}_m$ and thus $A(x) \overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \beta_1[\beta_2[\cdots[\beta_m]\cdots]]$. Therefore $A(x) \overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \alpha$.

We can prove the "if" part in a similar way. Since the basis of (1) and (2) and the induction step of (1) are almost same as the "only if" part, we will see only the induction step of (2). **Induction:** For $k \geq 2$, assume that the statement holds for any derivation of length less than k. (2) Suppose that $A(x) \overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \alpha$ is a derivation of length k. Then the following derivation is possible for some $m \geq 1$, $\beta_1, \beta_2, \ldots, \beta_m \in \mathrm{RHS}_{(\mathrm{iv})}$, $\gamma_1, \gamma_2, \ldots, \gamma_{m-1} \in T_{N_1}(X_1)$, and $\alpha_1, \alpha_2, \ldots, \alpha_m \in T_\Sigma(X_1)$ such that for $1 \leq i \leq m-2$, $\gamma_i \underset{\mathcal{G}'}{\Rightarrow} \beta_{i+1}[\gamma_{i+1}]$, $\gamma_{m-1} \underset{\mathcal{G}'}{\Rightarrow} \beta_m$, and for $1 \leq i \leq m$, $\beta_i \overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \alpha_i$.

$$
\begin{aligned}
A(x) &\underset{\mathcal{G}'}{\Rightarrow} \beta_1[\gamma_1] \\
&\underset{\mathcal{G}'}{\Rightarrow} \beta_1[\beta_2[\gamma_2]] \\
&\underset{\mathcal{G}'}{\Rightarrow} \beta_1[\beta_2[\beta_3[\gamma_3]]] \\
&\vdots \\
&\underset{\mathcal{G}'}{\Rightarrow} \beta_1[\beta_2[\cdots[\beta_{m-1}[\gamma_{m-1}]]\cdots]] \\
&\underset{\mathcal{G}'}{\Rightarrow} \beta_1[\beta_2[\cdots[\beta_{m-1}[\beta_m]]\cdots]] \\
&\overset{*}{\underset{\mathcal{G}'}{\Rightarrow}} \alpha_1[\alpha_2[\cdots[\alpha_{m-1}[\alpha_m]]\cdots]] = \alpha
\end{aligned}
$$

By the induction hypothesis, for $1 \leq i \leq m$, $\beta_i \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \alpha_i$. Because $A \overset{*}{\underset{\mathcal{G}_{\mathrm{gnf}}}{\Rightarrow}} \bar{\beta}_1\bar{\beta}_2\cdots\bar{\beta}_m$, $A \overset{*}{\underset{\mathcal{G}_{\mathrm{cnf}}}{\Rightarrow}} \bar{\beta}_1\bar{\beta}_2\cdots\bar{\beta}_m$ and thus $A(x) \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \beta_1[\beta_2[\cdots[\beta_m]\cdots]]$. Therefore $A(x) \overset{*}{\underset{\mathcal{G}}{\Rightarrow}} \alpha$.

Finally, we replace productions of the form $A \to B(C)$. Let $\hat{P}$ be the smallest set satisfying the following conditions:

- $P' \subseteq \hat{P}$.
- If $A \to B(C)$ is in P' and $B(x) \to \beta$ is in P', then $A \to \beta[C]$ is in $\hat{P}$.

Let P'' be the set obtained from $\hat{P}$ removing all productions of the form $A \to B(C)$. The LM-CFTG $\mathcal{G}'' = (N', \Sigma, P'', S)$ is clearly equivalent to $\mathcal{G}'$. It is clear that $\mathcal{G}''$ is in Greibach-like normal form and equivalent to $\mathcal{G}$. □

5. Conclusion

Chomsky-like normal form and Greibach-like normal form for LM-CFTGs were defined. It was shown that any LM-CFTG can be transformed into an equivalent one in both normal forms. These normal forms are helpful to develop and refine parsing algorithms [3] and lexicalization techniques [4] for LM-CFTGs.

Acknowledgment

The author would like to thank the anonymous referees for their helpful comments and suggestions. This study is supported in part by a Grant-in-Aid for Young Scientists ((B) 17700004) from the Japanese Ministry of Education, Culture, Sports, Science and Technology.

References

[1] A. Abeillé and O. Rambow, eds., Tree adjoining grammars: Formalisms, linguistic analysis and processing, CSLI Publications, Stanford, California, 2000.
[2] A. Fujiyoshi, "Linearity and nondeletion on monadic context-free tree grammars," Inf. Process. Lett., vol.93, no.3, pp.103–107, 2005.
[3] A. Fujiyoshi and I. Kawaharada, "Deterministic recognition of trees accepted by a linear pushdown tree automaton," Proc. 10th International Conference on Implementation and Application of Automata (CIAA2005), LNCS 3845, pp.129–140, Sophia Antipolis, 2005.
[4] A. Fujiyoshi, "Epsilon-free grammars and lexicalized grammars that generate the class of the mildly context-sensitive languages," Proc. 7th International Workshop on Tree Adjoining Grammar and Related Formalisms (TAG+7), pp.16–23, Vancouver, 2004.
[5] A. Fujiyoshi and T. Kasai, "Spinal-formed context-free tree grammars," Theory of Computing Systems, vol.33, no.1, pp.59–83, 2000.
[6] A.K. Joshi, L.S. Levy, and M. Takahashi, "Tree adjunct grammars," J. Comput. Syst. Sci., vol.10, no.1, pp.136–163, 1975.
[7] A.K. Joshi and Y. Schabes, "Tree-adjoining grammars," in Handbook of Formal Languages, ed. G. Rozenberg and A. Salomaa, pp.69–124, Springer-Verlag, Berlin, 1997.
[8] J.E. Hopcroft and J.D. Ullman, Introduction to Automata Theory, Languages and Computation, Addison Wesley, Reading, Massachusetts, 1979.
[9] S. Rajasekaran, "Tree-adjoining language parsing in $O(n^6)$ time," SIAM J. Comput., vol.25, no.4, pp.862–873, 1996.
[10] S. Rajasekaran and S. Yooseph, "TAL recognition in $O(M(n^2))$ time," J. Comput. Syst. Sci., vol.56, no.1, pp.83–89, 1998.
[11] W.C. Rounds, "Mapping and grammars on trees," Mathematical Systems Theory, vol.4, no.3, pp.257–287, 1970.
[12] K. Vijay-Shanker and D.J. Weir, "The equivalence of four extensions of context-free grammars," Mathematical Systems Theory, vol.27, no.6, pp.511–546, 1994.

Akio Fujiyoshi was born in Tokyo, Japan in 1971. He received the B.E., M.E., and Dr. Sci. degrees from the University of Electro-Communications, Tokyo, Japan, in 1995, 1997, and 2000, respectively. He is presently an assistant professor in the Department of Computer and Information Sciences, Ibaraki University. His main interests are formal language theory and algorithmic learning theory.