

CFGにおける並列性：多ヘッドCFG，部分同期CFG，および交代CFG

東京女子大学（文理） 守屋悦朗 （Etsuro Moriya）

1. はじめに

本稿ではCFG（文脈自由文法）における並列的導出法のいくつかについて考察する。それらは、（1）プロダクションの適用位置を指定する”書き換えヘッド”を複数個持つ多ヘッドCFG，（2）非終端記号を同期的なものと非同期的なものに分け、同期的な非終端記号については、導出途中の文形式において現れる全てに同一のプロダクションを適用して書き換えを行う部分同期CFG，（3）非終端記号を存在的なものと同期的なものに分け、alternationの概念[1, 2]を導入した交代CFG，の3つである。

本稿では[8]の結果を一部拡張し、いくつかの新しい結果を加えた。また、本稿の第4節は[9]でさらに詳細に扱われている。

2. 多ヘッドCFG

$G=(N, \Sigma, P, S)$ をCFGとする。kを正整数、Pを $P^k \times \mathbb{Z}^k$ の有限部分集合とすると、 $G=(N, \Sigma, P, S)$ をkヘッドCFGという。Pの元は

$$(*) \quad (X_1 \rightarrow x_1, \dots, X_k \rightarrow x_k, d_1, \dots, d_k)$$

という形の書き換え規則であり、 $(X_1 \rightarrow x_1, \dots, X_k \rightarrow x_k)$ はプロダクション部分、 $(d_1, \dots, d_k)$ はヘッド移動方向指定部分である。GをGの基礎CFGという。Gが λ -freeであるとはGが λ -freeのことであり、Gが1方向型であるとは、Pが $P^k \times \mathbb{N}^k$ の有限部分集合となっていることである。ただし、 $\mathbb{N} = \{0, 1, 2, \dots\}$ である。

$\{\Gamma_1, \dots, \Gamma_k\} \cap (N \cup \Sigma) = \emptyset$ とする（記号 $\Gamma_1, \dots, \Gamma_k$ はGの書き換えヘッドと呼ばれる）。Gにおける様相とは、各 $\Gamma_i (1 \leq i \leq k)$ を丁度1つずつ含んでいるような $(N \cup \Sigma \cup \{\Gamma_1, \dots, \Gamma_k\})^*$ の元のことをいう。様相a, ヘッド Γ_i に対して

$$a = b\Gamma_i c$$

とする。cの中の最左の非終端記号をXとすると、 Γ_i はXを指しているといい、cが非終端記号を含まないとき Γ_i は使われていないという。

多ヘッドCFGにおける導出を次のように定義する。様相aにおいて、全てのヘッドが使わ

れており、かつ、ヘッド $\Gamma_i (1 \leq i \leq k)$ は非終端記号 X_i を指しているとする。もし Γ_i と Γ_j が同じ非終端記号を指しているならば、すなわち、

$$a = a' \Gamma_i a'' \Gamma_j a''' \quad \text{かつ} \quad a'' \in (\Sigma \cup \{\Gamma_1, \dots, \Gamma_k\})^*$$

ならば、(*)のプロダクション部分の第 i プロダクション $A_i \rightarrow x_i$ と第 j プロダクション $A_j \rightarrow x_j$ とが一致する場合のみプロダクション(*)は a に適用でき、第 i ヘッド Γ_i が指している非終端記号は(*)の第 i プロダクションによって x_i に書き換えられる。この書き換えは全てのヘッドについて同時に行われる。書き換えを行った後、第 i ヘッドは自分の右側に在る d_i 個の非終端記号を読み飛ばすように右に移動する ($d_i < 0$ のときは、 $-d_i$ 個左に移動する)。この結果、 a から b が得られるとき

$$a \Rightarrow b$$

と書く。 $\Rightarrow^*$ を $\Rightarrow$ の反射推移閉包とする。

$$L(G) = \{w \in \Sigma^* \mid \Gamma_1 \dots \Gamma_k S \Rightarrow^* w_0 \Gamma_{i_1} w_1 \Gamma_{i_2} \dots w_{k-1} \Gamma_{i_k} w_k, w = w_0 w_1 \dots w_k, \\ \text{かつ } (i_1 i_2 \dots i_k) \text{ は } (1 2 \dots k) \text{ の置換}\}$$

を G が生成する k ヘッド CFL と呼ぶ。

多ヘッド CFG は、simple matrix 文法 [6]，matrix 文法 [10]，scattered context 文法 [5] などの自然な拡張になっていることに注意する。(matrix 文法のプロダクション $[X_1 \rightarrow x_1, \dots, X_n \rightarrow x_n]$ においては、どの X_i も $x_1, \dots, x_{i-1}$ の中に現れないと仮定することができるので、ヘッド移動用のプロダクションを用意しておけば、この matrix プロダクションは n 個のヘッドでシミュレートできる。)

定理 1. 1 方向型 2 ヘッド CFL のクラスは帰納的可算集合のクラスと一致する。実際、基礎 CFG を線形文法としてこのことがいえるが、基礎 CFG を正規文法とすることは出来ない。

証明の方針 任意の帰納的可算集合は 2 つの線形 CFL の intersection の homomorphic image として表される [3] ことを用いる。2 つの CFL の intersection は、2 つの Greibach 標準形の CFG を 2 つのヘッドによって同時にシミュレートする。

定理 2. 任意の k に対して、2 方向型 λ -free k ヘッド CFL は文脈依存言語である。(2 方向型 λ -free 多ヘッド CFG によって生成されえないような文脈依存言語が存在するか否かはわかっていない。)

3. 部分同期 CFG

$G = (N, \Sigma, P, S)$ を CFG とする。非終端記号を同期的非終端記号と非同期的非終端記号に

分ける：

$$N = N_s \cup N_a, \quad N_s \cap N_a = \emptyset$$

左辺が同期的非終端記号 (N_s の元) であるプロダクションを同期プロダクション, そうでないものを非同期プロダクションという. $G = (N_a, N_s, \Sigma, P, S)$ を部分同期CFGという.

非同期プロダクションは, 通常のCFGのプロダクションと同様に適用され導出が行われる. この導出 (非同期導出) を $\Rightarrow_a$ で表す. すなわち, $a \Rightarrow_a b$ となるのは, $a = a'Xa''$, $b = a'xa''$ となる非同期プロダクション $X \rightarrow x$ が存在するときである. これに対し, 同期プロダクションは次のように”同期的に”適用され, 導出が行われる (同期導出).

$$a = a_0X_1a_1\dots X_na_n, \quad a_i \in (N_a \cup \Sigma)^* \quad (0 \leq i \leq n)$$

とし, 各 $X_i \rightarrow x_i$ ($1 \leq i \leq n$) を同期プロダクションとする.

$$b = a_0x_1a_1\dots x_n a_n$$

かつ

$$X_i = X_j \quad \text{ならば} \quad x_i = x_j$$

であるとき, $a \Rightarrow_s b$ と書く.

”同期導出”は次のように定義することも考えられる. すなわち, $a = a_0Aa_1\dots Aa_n$ で, どの a_i ($0 \leq i \leq n$) も A を含まないとするとき, $A \rightarrow x$ が同期プロダクションで $b = a_0xa_1\dots xa_n$ であるならば $a \Rightarrow_t b$ であると定義する. しかし, 次の補題に示すように, $\Rightarrow_s$ と $\Rightarrow_t$ とは本質的に同じものである.

補題. 同期導出 $a \Rightarrow_t b$ が成り立つ必要十分条件は, 同期導出 $a \Rightarrow_s t$ が成り立つことである.

$a \Rightarrow_a b$ または $a \Rightarrow_s b$ であるとき

$$a \Rightarrow b$$

と書く.

$$L(G) = \{ w \in \Sigma^* \mid S \Rightarrow^* w \}$$

を G により生成された部分同期CFLと呼ぶ.

定理3. L が部分同期CFLならば, $L - \{\lambda\}$ は文脈依存言語である. また, 部分同期CFLでない文脈依存言語が存在する.

部分同期CFGにおける交代導出を次のように定義する.

$$(*) \quad a = a_0 \Rightarrow^{*q_1} a_1 \Rightarrow^{*q_2} \dots \Rightarrow^{*q_n} a_n = b$$

とする。ただし、どの i についても q_i は s または a であり、かつ、 $q_i \neq q_{i+1}$ とする。また、 $\Rightarrow^{*q_i}$ は $\Rightarrow_{q_i}$ の推移閉包である。もし、 $q_1 = a$ ならば $(**)$ を Σ_n -導出、 $q_1 = s$ ならば $(**)$ を Π_n -導出と呼ぶ。

任意の $w \in L(G)$ に対して w を生成する Σ_m -導出 ($m \leq n$) が存在するならば G は Σ_n -PSCFG と呼ばれ、 Σ_n -PSCFG によって生成される言語を Σ_n -PSCFL と呼ぶ。 Π_n -PSCFG, Π_n -PSCFL も同様に定義される。

Σ_n -PSCFL, Π_n -PSCFL のクラスをそれぞれ

$$\Sigma_n^{\text{PSCF}}, \quad \Pi_n^{\text{PSCF}}$$

で表す。定義から、 Σ_1^{PSCF} は CFL のクラスそのものである。

定理4 [11]. (1) Σ_1^{PSCF} と Π_1^{PSCF} は比較不能である。

(2) $\Sigma_1^{\text{PSCF}} \cap \Pi_1^{\text{PSCF}}$ は derivation bounded 言語 [4] のクラスと一致する。

$$\Sigma_n^{\text{PSCF}} \cup \Pi_n^{\text{PSCF}} \subseteq \Sigma_{n+1}^{\text{PSCF}} \cap \Pi_{n+1}^{\text{PSCF}}$$

が成り立つことは容易に証明できる。また、 $n=1$ のときこれは真の包含関係であるが、一般の n についてはまだ証明されていない。

$\mathcal{I}_1, \mathcal{I}_2$ を言語のクラスとすると、

$$\mathcal{I}_1 \sigma \mathcal{I}_2 = \{ \tau(L_1) \mid L_1 \subseteq \Sigma^* \text{ は } \mathcal{I}_1 \text{ の元, かつ, } \tau \text{ はどの } a \in \Sigma \text{ についても } \tau(a) \in \mathcal{I}_2 \text{ であるような代入} \}$$

$$\mathcal{I}_1 \pi \mathcal{I}_2 = \{ \mathcal{R}(L_1) \mid L_1 \subseteq \Sigma^* \text{ は } \mathcal{I}_1 \text{ の元, かつ, } \mathcal{R} \text{ はどの } a \in \Sigma \text{ についても } \mathcal{R}(a) \in \mathcal{I}_2 \text{ であるような準同型写像のクラス} \}$$

と定義する。ただし、 $L \subseteq \Sigma^*$ に対して、

$$\mathcal{R}(L) = \{ h(L) \mid h \in \mathcal{R} \}$$

である。とくに、言語のクラス $\mathcal{I}$ に対して、

$$\sigma_1(\mathcal{I}) = \mathcal{I} \sigma \Sigma_1^{\text{PSCF}}, \quad \pi_1(\mathcal{I}) = \mathcal{I} \pi \Pi_1^{\text{PSCF}}$$

と定義し、 $\mathcal{I}$ を含み σ_1 および π_1 で閉じている最小のクラスを $\mathcal{I}$ の (σ_1, π_1) 閉包という。

定理5. n を任意の自然数とする.

(1) n が偶数ならば,

$$\Sigma_{n+1}^{\text{PSCF}} = \sigma_1(\Sigma_n^{\text{PSCF}}), \quad \Pi_{n+1}^{\text{PSCF}} = \pi_1(\Pi_n^{\text{PSCF}}).$$

(2) n が奇数ならば,

$$\Sigma_{n+1}^{\text{PSCF}} = \pi_1(\Sigma_n^{\text{PSCF}}), \quad \Pi_{n+1}^{\text{PSCF}} = \sigma_1(\Pi_n^{\text{PSCF}}).$$

系. 部分同期CFLのクラスは, 有限言語のクラスの (σ_1, π_1) -閉包に一致する.

4. 交代CFG

交代CFGとは5つ組 $G = (N, U, \Sigma, P, S)$ のことをいう. ここに, (N, Σ, P, S) はCFGであり, U は N の部分集合である. N の元を存在的非終端記号, U の元を全称的非終端記号という.

G における導出を次のように定義する. T を各頂点に $(N \cup \Sigma)^*$ の元がラベル付けされた木とする. 以下, 頂点 π のラベルを $\iota(\pi)$ で表すことにする. T のどの内部頂点 π についても次の(i)または(ii)が成り立っているとする.

(i) $\iota(\pi) = yxz$, $x \in N \cup U$, $x \rightarrow x \in P$, かつ, π は丁度1つの子供 ρ を持ち, $\iota(\rho) = yxz$ である. π を存在的頂点と呼ぶ.

(ii) $\iota(\pi) = yxz$, $x \in U$, P の x -プロダクション(左辺が x であるようなプロダクションのこと)は $x \rightarrow x_1 | x_2 | \dots | x_k$, かつ, π は丁度 k 個の子供 $\rho_1, \dots, \rho_k$ を持ち, $\iota(\rho_i) = yx_i z$ である. π を全称的頂点と呼ぶ.

このとき, T を G における $\iota(T$ の根)からの導出という. 導出 T の各頂点 π の値 $\text{value}(\pi)$ を次のように定義する.

$$\text{value}(\pi) = \begin{cases} \iota(\pi) & \pi \text{が} T \text{の葉であるとき} \\ \text{value}(\rho) & \pi \text{が存在的頂点で} \rho \text{がその子供のとき} \\ w & \pi \text{が全称的頂点で, そのどの子供} \rho \text{についても} \\ & \text{value}(\rho) = w \text{であるとき} \\ \text{定義されない} & \text{その他} \end{cases}$$

T の根の値を T の値と定義し, やはり $\text{value}(T)$ と表すことにする. $\text{value}(T)$ が定義されている導出のことを正当な導出と呼ぶ.

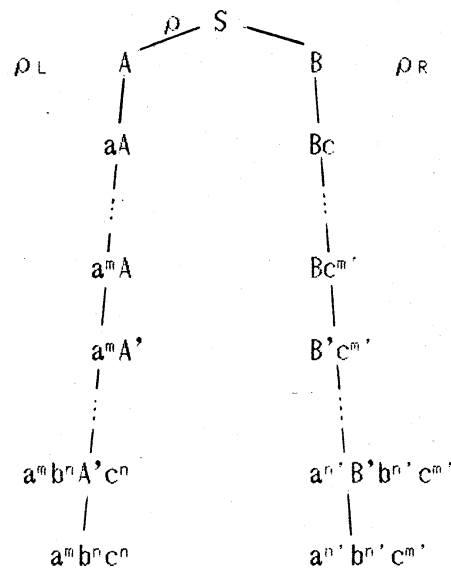
$$L(G) = \{ \text{value}(T) \in \Sigma^* \mid T \text{ は } G \text{ における } S \text{ からの正当な導出} \}$$

を G によって生成された交代CFLという.

例 交代CFG $G = (\{S, A, A', B, B'\}, \{S\}, \{a, b, c\}, P, S)$ を考える. 但し, P は次のプロダクションから成るとする:

$$\begin{aligned} S &\rightarrow A \mid B \\ A &\rightarrow aA \mid A', \quad A' \rightarrow bA'c \mid \lambda \\ B &\rightarrow Bc \mid B', \quad B' \rightarrow aB'b \mid \lambda \end{aligned}$$

G における導出のうちで, $\text{value}(T)$ が終端記号列となるものは必ず次の形をしている:



$\text{value}(\rho_L) = a^m b^n c^n$, $\text{value}(\rho_R) = a^{n'} b^{n'} c^{m'}$ であり, S は全稱的非終端記号だから, $\text{value}(\rho)$ が定義され T が正当な導出となるためには, $m=n=n'=m'$ となる必要がある. よって, $L(G) = \{ a^n b^n c^n \mid n \geq 0 \}$ である.

交代CFGと交代プッシュダウンオートマトン(alternating pda)とは, 次のような深い関係にある.

定理6. 交代CFG G に対して $L=L(G) \iff$ 交代プッシュダウンオートマトン M に対して $L=L(M)$.

証明の方針 $CFG \iff PDA$ の証明をほぼなぞる. 交代CFGにおける最左導出とは, 根から葉へ至る全ての道がCFGにおける導出とみたとき最左導出となっているような導出のことであると定義する. 交代CFG G においても, 任意の $w \in L(G)$ に対して, w を生成する最左導出が存在することがいえる. このことに注意すれば, $CFG \iff PDA$ の証明と同様に, 交代CFGにおける最左導出と交代プッシュダウンオートマトンにおけるID (入力とスタックの内容) とをうまく対応させることが出来る.

交代オートマトンと同様に、交代CFGにおける導出でも、全称的導出部分と存在的導出部分との交代回数で導出を分類することが出来る。

G を交代CFG, k を正整数とする。 G における導出 T が Σ_k -導出(Π_k -導出)といわれるのは、 T の根が存在的頂点(全称的頂点)であり、かつ、 T における根から葉へ至るどの道においても存在的頂点/全称的頂点から全称的頂点/存在的頂点への交代が高々 $k-1$ 回しか起こっていないときである。任意の $w \in L(G)$ に対して、 w を生成する Σ_k -導出が存在するとき G は Σ_k -交代CFGといい、 $L(G)$ を Σ_k -交代CFLという。 Π_k -交代CFG, Π_k -交代CFLも同様に定義される。 Σ_k -交代CFLのクラス、 Π_k -交代CFLのクラスをそれぞれ

$$\Sigma_k^{ACF}, \quad \Pi_k^{ACF}$$

で表す。次の関係が成り立つことは容易に示される：

$$\Sigma_n^{ACF} \cup \Pi_n^{ACF} \subseteq \Sigma_{n+1}^{ACF} \cap \Pi_{n+1}^{ACF} \quad (n \geq 1).$$

定理7. $\Pi_1^{ACF} \subseteq \Sigma_2^{ACF} \subseteq \Pi_3^{ACF} \subseteq \dots \subseteq \Sigma_{2n}^{ACF} \subseteq \Pi_{2n+1}^{ACF} \subseteq \Sigma_{2n+2}^{ACF} \subseteq \dots$

$$\begin{array}{ccccccc} \parallel & & \parallel & & \parallel & & \parallel \\ \Sigma_1^{ACF} & \subseteq & \Pi_2^{ACF} & \subseteq & \dots & \subseteq & \Sigma_{2n-1}^{ACF} \subseteq \Pi_{2n}^{ACF} \subseteq \Sigma_{2n+1}^{ACF} \subseteq \dots \end{array}$$

特に、 Π_1^{ACF} は空集合およびsingletonから成るクラスである。

参考文献

1. A.K.Chandra, D.C.Kozen and L.J.Stockmeyer, Alternation, JACM 28 (1981), 114-133.
2. A.K.Chandra and L.J.Stockmeyer, Alternation, Proc. 17th FOCS (1976), 88-108.
3. S.Ginsburg, S.Greibach and M.Harrison, One-way stack automata, JACM 14 (1967), 389-418.
4. S.Ginsburg and E.H.Spanier, Derivation bounded languages, JCSS 2 (1968), 228-250.
5. S.A.Greibach and J.E.Hopcroft, Scattered context grammars, JCSS 3 (1969), 233-247.
6. O.H.Ibarra, Simple matrix languages, Inform. Contr. 17 (1970), 359-394.
7. R.Ladner, R.J.Lipton and L.J.Stockmeyer, Alternating pushdown and stack automata, SIAM J. Comput. 13 (1984), 135-155.
8. E.Moriya, On parallelism in context-free grammars, to appear in North-Holland Mathematics Series.
9. E.Moriya, A grammatical characterization of alternating pushdown automata, in preparation.
10. A.Salomaa, Formal Languages, Academic Press, New York, 1973.
11. P.Siromoney and K.Kriithivasan, Parallel context-free languages, Inform. Contr. 24 (1974), 155-162.