

有界交互文脈自由言語について

電気通信大学 中村 愛 (Megumu Nakamura)

1. まえがき

交互性(Alternation)の概念は非決定性の一般化として導入された。[1]に交互を起こすブッシュダウン・オートマトンについての研究がある。交互性を文脈自由言語にはじめて持ち込んだのは[2]である。[2]では交互文脈自由言語のクラス(ALT-CFL)と交互ブッシュダウン・オートマトンの受理する言語のクラス(ALT-PDA)が等しいことを示している。交互ブッシュダウン・オートマトンについては[1]で $U_{\text{CFL}} \text{DTIME}(C^n) = \text{ALT-PDA}$ が示されており[1]、[2]より交互文脈自由言語のクラスがチューリング機械でキャラクタライズできる自然なクラスであることがわかる。また交互の回数を制限した交互ブッシュダウン・オートマトンの受理する言語のクラス(BND-APDA)については[5]で $\text{NSPACE}(n) \subseteq \text{BND-APDA} \subseteq \text{DSpace}(n^2)$ が示されている。

本稿では交互の回数が有限回に制限された交互文脈自由文法(有界交互文脈自由文法と呼ぶ)を定義する。またこの文法の生成する言語のクラス(BND-ACFL)は文脈自由言語を含み演算 U , $\cap$, 連接の下で閉じている十分大きく自然なクラスであることを示す。特に共通部分 $\cap$ の下で閉じていることはこのクラスの特徴である。次に、文脈自由言語と同様、任意の有界交互文脈自由言語は $A \rightarrow \lambda$ という形の規則(これを λ -規則と呼ぶ)を用いずに生成できることを示す。これを用いて本稿の主定理である任意の有界交互文脈自由言語が $\Theta(n)$ 領域限定決定性チューリン

グ機械で受理できること($\text{BND-ACFL} \subseteq \text{DSPACE}(n)$)が容易に示される。

一方、交互文脈自由文法についてはこのような λ -規則の除去が不可能ではないかと予想される。なぜならもし λ -規則の除去が可能なら同様の方法を用いて $\text{ALT-PDA} \subseteq \text{DSPACE}(n)$ となり $U_{C \rightarrow a} \text{DTIME}(C^n) \subseteq \text{DSPACE}(n)$ がいえてしまうからである。またもし $\text{BND-ACFL} = \text{BND-APDA}$ なら $\text{DSPACE}(n) = \text{NSPACE}(n)$ となる。以上の結果をまとめると $\text{BND-ACFL} \subseteq \text{DSPACE}(n) \subseteq \text{NSPACE}(n) \subseteq \text{BND-APDA} \subseteq \text{DSPACE}(n^2)$ となる。

2. 準備

この章では本稿で扱う文法と用語の定義を行う。

交互文脈自由文法 (Alternating Context-Free Grammar, ACFG と略す) とは5つ組 $G = (N, U, \Sigma, P, S)$ のことをいう。ここで (N, Σ, P, S) は文脈自由文法 (Context-Free Grammar, CFG と略す) で、それぞれ変数の集合、終端記号の集合、規則の集合、および開始記号を表す。 U は N の部分集合で、 $E = N - U$ の元を存在変数、 U の元を全称変数という。 $A \in N$ に対して左辺が A である規則を A -規則という。

$w \in \Sigma^*$ 、 $\beta \in (N \cup \Sigma)^*$ で規則 $A \rightarrow \alpha$ が P にあるとき P によって $wA\beta$ から $w\alpha\beta$ が直接生成されるといい $wA\beta \Rightarrow w\alpha\beta$ であらわす。関係 $\Rightarrow$ の反射・推移閉包を $\twoheadrightarrow$ であらわす。 $\alpha \twoheadrightarrow \beta$ のとき P によって α から β が生成されるという。 P を省略して $\Rightarrow$ 、 $\twoheadrightarrow$ と書くこともある。

ACFG の動作は他の交互モデルと同様に有限の根付き木を用いて定義される。

$G = (N, U, \Sigma, P, S)$ を ACFG とし、 $E = N - U$ とおく、このとき G における語 $w \in \Sigma^*$ の追跡木とは $(N \cup \Sigma)^*$ の元が各頂点にラベルづけされた有限の根付き木で、次の条件 (I)、(II)、(III) を満たすものである。

(I)、木の根のラベルはSである

(II)、木の葉のラベルはすべてwである

(III)、 π を葉でない頂点、そのラベルを $x A \beta$ ($x \in \Sigma^*$ 、 $A \in N$ 、 $\beta \in (N \cup \Sigma)^*$)とする。 $A \rightarrow \alpha_1 | \alpha_2 | \dots | \alpha_k$ がPにあるすべてのA-規則であるとき

(a)、 $A \in E$ なら、 π は1つの子を持ち、そのラベルは $x \alpha_1 \beta$ 、 $x \alpha_2 \beta$ 、 $\dots$ 、 $x \alpha_k \beta$ のうちのいずれかである。このとき π は存在的頂点であるという。

(b)、 $A \in U$ なら、 π はk個の子供を持ち、それらのラベルは $x \alpha_1 \beta$ 、 $x \alpha_2 \beta$ 、 $\dots$ 、 $x \alpha_k \beta$ である。このとき π は全称的頂点であるという。

条件(I)のSを任意の変数Aで置き換えて定義される木をA-追跡木と呼ぶ。

$L(G) = \{w \mid \text{ACFG } G \text{ による } w \text{ の追跡木が存在する} \}$ をGが生成する交互文脈自由言語(Alternating Context-Free Language, ACFLと略す)という。

追跡木の根から葉への道の上の存在的(全称的)頂点から全称的(存在的)頂点への移行を交互(Alternation)という。根から葉までの道上の交互の回数の最大値を追跡木の交互の回数とする。

定義 ACFG $G = (N, U, \Sigma, P, S)$ のどの追跡木も交互の回数が高々 $k-1$ 回のときGはk有界であるという。交互の回数が単に有限回であるとき有界であるという。(k)有界ACFGで生成される言語を(k)有界ACFLという。

最後に本稿で扱う計算量のクラスの定義を行う。

ALT-CFL: 交互文脈自由文法によって生成される言語のクラス

ALT-PDA: 交互プッシュダウン・オートマトンによって受理される言語のクラス

BND-ACFL: 交互の回数が限定された交互文脈自由文法によって生成される言語のクラス

BND-APDA: 交互の回数が限定された交互プッシュダウン・オートマトンによって受理される言語のクラス

DSPACE($f(n)$): 決定性チューリング機械によって $f(n)$ 領域で受理される言語のクラス

NSPACE($f(n)$): 非決定性チューリング機械によって $f(n)$ 領域で受理される言語のクラス

3. ACFLと有界ACFL

CFLとPDAの受理する言語のクラスは等しく、またACFLとAPDAはCFLとPDAにそれぞれ交互性を導入したものである。この2つの関係については次の結果が得られている。

定理 (守屋)

$$\text{ALT-CFL} = \text{ALT-PDA}$$

有界ACFLはACFLの部分集合であるがCFLを自然に拡張した言語のクラスを含む充分大きなクラスであることを示す。

定理 1

有界ACFLのクラスは $\cup$, $\cap$, 連接の演算のもとで閉じている

証明略

よって、次のことが容易にわかる。

系 1

有界ACFLのクラスはCFLに演算 $\cap$, $\cup$, $*$, 接続を有限回ほどこして得られる言語のクラスを含む。

4. 有界ACFGの簡単化

この章では文脈自由文法と同様に有界交互文脈自由文法においても $A \rightarrow \lambda$ という形の規則が除去できることを示す。この結果を用いて次の章で有界ACFLのクラスについて考える。

定義 $A \rightarrow \lambda$ の形の規則を λ -規則という。

定義 有界ACFG $G = (N, U, \Sigma, P, S)$ において $\alpha \in (N \cup \Sigma)^*$ が G による追跡木のラベルとなるなら α は G の生成語であるという。

変数 A, B がある $\alpha, \beta \in (N \cup \Sigma)^*$, $C \in N$ に対して①, ②, ③のいずれかのとき A は B より先に現れるという。

- ① 規則 $A \rightarrow \alpha B \beta$ が P にある。
- ② 規則 $C \rightarrow \alpha B \beta$ が P にありかつ $\alpha \neq \alpha' A \beta'$
- ③ A が C より先に現れ C が B より先に現れる

補題 1

任意の有界ACFLは次の条件を満たす有界ACFG $G = (N, U, \Sigma, P, S)$ で生成できる

条件 $A \in U$, $A \rightarrow \alpha \in P$ なら $\alpha \in E$

証明略

次に有界ACFGの標準形を示す

補題2

ACFG $G = (N, U, \Sigma, P, S)$, $E = N - U$ が k 有界であるとき $L(G)$ を生成する ACFG で 1), 2), 3) を満たすものが存在する

- 1) $N = \cup N_i$ ただし全ての i に対して $N_i \subseteq U$ または $N_i \subseteq E$ 、かつ $i \neq j$ なら $N_i \cap N_j = \emptyset$
- 2) $N_i \subseteq E$ なら $N_{i+1} \subseteq U$ かつ $N_i \subseteq U$ なら $N_{i+1} \subseteq E$ ただし $1 \leq i \leq k-1$
- 3) $A \in N_i, B \in N_j$ かつ A が B より先に現れるなら $i \geq j$

証明略

以降の有界 ACFG は補題 1, 2 を満たす形で与えられることとする。

定義 有界 ACFG $G = (N, U, \Sigma, P, S)$ で $A, X \in N$, $\gamma \in (N \cup \Sigma)^*$ に対して

$$A \xrightarrow{\gamma} X$$

なら A は X の最右生成可能であるという。

$$N_X = \{A \mid A \xrightarrow{\gamma} X\} \text{ とおく。}$$

補題3

任意の有界 ACFG $G = (N, U, \Sigma, P, S)$, $N = \cup N_i$ において $X \rightarrow \lambda \in P$ なら G の生成語 α は

$$\bar{N}_X^* N_X \bar{N}_X^* X \bar{N}_X^* \cup \bar{N}_X^* N_X \bar{N}_X^* \cup \bar{N}_X^*$$

の元である。ただし $\bar{N}_X = (N - N_X) \cup \Sigma$

証明略

有界 ACFG における λ -規則の除去は、 X が λ -規則をもつとき、追跡木中の生成語において、変数 X の右側にあらわれる記号を X から直接生成するように規則を変形しておこなう。 X

から規則を右辺から左辺へ N_X が続く限りたどり生成語において X の右側に現れる記号を見つけることができる。

集合 Ω を次のように定義する。 λ -規則を持つ任意の X に対して

$$\Omega = \widetilde{N}_X^* N_X \widetilde{N}_X^* X \widetilde{N}_X^* \cup \widetilde{N}_X^* N_X \widetilde{N}_X^* \cup \widetilde{N}_X^*$$

とする

補題4

任意の ACFL L_1 と特別な記号 $\$$ に対し $L = L_1 \$$ を生成する有界 ACFG G が存在し、 G の任意の生成語は Ω の元である。

証明 L_1 を生成する有界 ACFG $G_1 = (N, U, \Sigma, P, S)$ において $X \rightarrow \lambda \in P$ なら、 L_1 の生成語 α は $\widetilde{N}_X^* N_X \widetilde{N}_X^* X \widetilde{N}_X^* \cup \widetilde{N}_X^* (N_X \cup \{\lambda\}) \widetilde{N}_X^*$ の元である。よって $G = (N \cup \{S'\}, U, \Sigma \cup \{\$, \}, P', S')$

$P' = P \cup \{S' \rightarrow S \$\}$ で与えられる。 $\square$

有界 ACFG は λ -規則を含むループを作らなくできるのでの最右生成可能変数を有限個除去することによって λ -規則の除去ができる。

定理2

任意の有界 ACFL L_1 に対し $L = L_1 \$$ は λ -規則を持たない有界 ACFG で生成される

証明 L を生成する有界 ACFG で補題4を満たすものを $G = (N, U, \Sigma, P, S)$, $N = \cup N_i$ とする。

G が m 個の λ -規則を持つとする次の操作で補題4を満たし $m-1$ 個の λ -規則を持つ有界 ACFG $G' = (N', U', \Sigma', P', S')$ で $L(G) = L(G')$ となるものを構成する

Ω の元 α を $\alpha = \alpha_1 \alpha_2 \cdots \alpha_k$, $\alpha_i \in N_X X \widetilde{N}_X \cup N_X \widetilde{N}_X \cup \widetilde{N}_X$ に分割する。この分割は任意の α

$\in \Omega$ に対して一意的に行える。これを標準分割と呼ぶ。

G' は次の様に構成する

$$I) N' = \cup N_i', N_i' = \{A_\beta \mid A \in N_i, \beta \in \{\lambda\} \cup \tilde{N}_X \cup X \tilde{N}_X\}$$

$$II) \Sigma' = \Sigma, S' = S_\lambda$$

P' を定めるために、 Ω の元 $\alpha \in (N \cup \Sigma)^*$ に対応する $\bar{\alpha} \in (N' \cup \Sigma')^*$ を次で定義する

α の標準分割 $\alpha_1 \alpha_2 \cdots \alpha_k$ に対し、 $\bar{\alpha} = \bar{\alpha}_1 \bar{\alpha}_2 \cdots \bar{\alpha}_k$ とする。ただし $A \in N_X, Z \in \tilde{N}_X, Y \in$

$N - N_X$ に対して

$$\alpha_i = A X Z \text{ なら } \bar{\alpha}_i = A_{XZ}$$

$$\alpha_i = A Z \text{ なら } \bar{\alpha}_i = A_Z$$

$$\alpha_i = Y \text{ なら } \bar{\alpha}_i = Y_\lambda$$

$$\alpha_i \in \Sigma^+ \text{ なら } \bar{\alpha}_i = \alpha_i$$

である。 α と $\bar{\alpha}$ は一対一に対応する。

$$III) P' = \{A_\beta \rightarrow \bar{\alpha} \beta \mid A \rightarrow \alpha \in P, \alpha \beta \in \Omega\}$$

P' は X_λ を左辺に持つ規則を含まないので λ -規則の数は $m-1$ 個となる。 $L(G) = L(G')$ の証明は略す。また、 $\bar{\alpha}$ の定義より G の生成語は必ず $\$, A_\$,$ または $A_{X\$}$ ($A \in N - N_X$) を右端に持つ。これらは任意の X ($X \rightarrow \lambda \in P$) に対しての最右生成可能ではないので、 G' は補題4を満たす。

この操作を高々 m 回繰り返して λ -規則を持たない有界 ACFG を構成できる。☒

定理3

任意の有界 ACFL L に対し、 $L - \{\lambda\}$ を生成する λ -規則を持たない有界 ACFG G が存在する

証明略

5. 結び

定理3より本稿の主定理が容易に導かれる

定理4

$$\text{BND-ACFL} \subseteq \text{DSPACE}(n)$$

また初期状態が存在的で、交互を $k-1$ 回以下しかおこさない 2-way プッシュダウン・

オートマトンによって受理される言語のクラス $\Sigma_k\text{-PDA}$ については

定理 (Ladner, Stockmeyer and Lipton)

$$\text{NSPACE}(n) \subseteq \Sigma_k\text{-PDA} \subseteq \text{DSPACE}(n^2)$$

が示されているので、次の系が得られる。

系2

もし $\text{BND-ACFL} = \text{BND-APDA}$ なら

$$\text{DSPACE}(n) = \text{NSPACE}(n)$$

本稿では交互の回数を限定した交互文脈自由言語—有回交互文脈自由言語を定義し、そのクラスと他のクラスの包含関係を示した。特に3章ではこの言語がCFLに演算 $\cup$ 、 $\cap$ 、 $*$ 、連接を有限回ほどこして得られるクラスを含むことを示した。3章以降の結果は次の通りである。

$$\begin{array}{ccccc}
 \text{PDA} & \subseteq & \text{BND-APDA} & \subseteq & \text{ALT-PDA} \\
 & & \cup & & \parallel \\
 & & \text{NSPACE}(n) & & \\
 \parallel & & \cup & & \bigcup_{c \geq 0} \text{DSPACE}(C^n) \\
 & & \text{DSPACE}(n) & & \parallel \\
 & & \cup & & \\
 \text{CFL} & = & \text{BND-ACFL} & \subsetneq & \text{ALT-CFL}
 \end{array}$$

謝辞 ご指導いただいた笠井助教授に深く感謝いたします。

Reference

- [1] A.K.Chandra, D.C.Kozen and L.J.Stockmeyer, Alternation, J.A.C.M.28 (1981)114-133.
- [2] 守屋悦朗、CFGにおける併列性—とくにalternationについて—、L.A.シンポジウム1987.2
- [3] J.E.Hopcroft and J.D.Ullman, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, Reading, MA, 1979
- [4] R.Ladner, R.J.Lipton and L.J.Stockmeyer, Alternating pushdown and stack automata, SIAM J.C. 13(1984) 135-155
- [5] R. Ladner, L.J.Stockmeyer and R.J.Lipton, Alternation Bounded Auxiliary Pushdown Automata I&C 62 (1984) 93-108
- [6] 中村愛、Alternating Context-free Grammar について、信学技報 Vol.87 No.77, 47-52