数理解析研究所講究録
第666巻 1988年 205-214

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］に交互を起こすプッシュダウン・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)と交互プッシュダウン・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。交互プッシュダウン・オートマトンについては［1］で$\bigcup_{c>0}$DTIME(C^n)=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プッシュダウン・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NSPACE(n)$\subseteq$BND-APDA$\subseteq$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$\cup$，$\cap$，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$A\rightarrow\lambda$と言う形の規則（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Θ(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$\cup_{c>0}\text{DTIME}(C^n) \subseteq \text{DSPACE}(n)$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、$E=N-U$の元を存在的変数、Ｕの元を全称的変数という。$A \in N$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$A \to \alpha$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$wA\beta \underset{P}{\Rightarrow} w\alpha\beta$であらわす。関係$\underset{P}{\Rightarrow}$の反射・推移閉包を$\overset{*}{\underset{P}{\Rightarrow}}$であらわす。$\alpha \overset{*}{\underset{P}{\Rightarrow}} \beta$のときＰによって$\alpha$から$\beta$が生成されるという。Ｐを省略して$\Rightarrow$、$\overset{*}{\Rightarrow}$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$xA\beta$（$x\in\Sigma^*$、$A\in N$、$\beta\in(N\cup\Sigma)^*$）とする。$A\to\alpha_1|\alpha_2|\cdots|\alpha_k$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$x\alpha_1\beta$、$x\alpha_2\beta$、…、$x\alpha_k\beta$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 $x\alpha_1\beta$、$x\alpha_2\beta$、…、$x\alpha_k\beta$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。

$L(G)=\{w\mid$ ACFG Gによるｗ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）

ＡＬＴ－ＣＦＬ＝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∪，∩，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∩，U，*，連接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$A \to \lambda$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　$A \to \lambda$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$\alpha$はGの生成語であるという。

変数A，Bがあるα、$\beta \in (N \cup \Sigma)^*$、$C \in N$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$A \to \alpha B \beta$がPにある。

② 規則$C \to \alpha B \beta$がPにありかつ$\alpha \overset{*}{\Rightarrow} \alpha' A \beta'$

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　$A \in U$，$A \to \alpha \in P$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$N_i \cap N_j \neq \phi$

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$1 \leqq i \leqq k-1$

３）　$A \in N_i$，$B \in N_j$かつＡがＢより先に現れるなら$i \geqq j$

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

$$A \overset{*}{\underset{P}{\Rightarrow}} \gamma X$$

ならＡはＸの最右生成可能であるという。

$N_X=\{A \mid A \overset{*}{\underset{P}{\Rightarrow}} \gamma X\}$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$X \to \lambda \in P$ならＧの生成語αは

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

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

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

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

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$\overline{N}_X{}^* N_X \overline{N}_X{}^* X \overline{N}_X{}^* \cup \overline{N}_X{}^* (N_X \cup \{\lambda\}) \overline{N}_X{}^*$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。　⊠

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$\alpha=\alpha_1\alpha_2\cdots\alpha_k$、$\alpha_i\in N_X X \overline{N}_X \cup N_X \overline{N}_X \cup \overline{N}_X$に分割する。この分割は任意の$\alpha$

$\in \Omega$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) $N' = \cup N_i'$、$N_i' = \{A_\beta \mid A \in N_i,\ \beta \in \{\lambda\} \cup \widetilde{N}_X \cup X\widetilde{N}_X\}$

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$\overline{\alpha} \in (N' \cup \Sigma')^*$を次で定義する

αの標準分割$\alpha_1 \alpha_2 \cdots \alpha_k$に対し、$\overline{\alpha} = \overline{\alpha_1}\,\overline{\alpha_2} \cdots \overline{\alpha_k}$とする。ただし$A \in N_X$、$Z \in \widetilde{N}_X$、$Y \in N - N_X$に対して

$\alpha_i = AXZ$なら$\overline{\alpha_i} = A_{XZ}$

$\alpha_i = AZ$なら$\overline{\alpha_i} = A_Z$

$\alpha_i = Y$なら$\overline{\alpha_i} = Y_\lambda$

$\alpha_i \in \Sigma^+$なら$\overline{\alpha_i} = \alpha_i$

である。αと$\overline{\alpha}$は一対一に対応する。

III) $P' = \{A_\beta \to \overline{\alpha\beta} \mid A \to \alpha \in P、\alpha\beta \in \Omega\}$

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、$\overline{\alpha}$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($X \to \lambda \in P$)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－{λ}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

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

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Σ_k－ＰＤＡについては

定理（Ladner, Stockmeyer and Lipton）

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

が示されているので、次の系が得られる。

系２

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

$$DSPACE(n) = NSPACE(n)$$

本稿では交互の回数を限定した交互文脈自由言語一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∪、∩、＊、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。

$$
\begin{array}{ccccc}
\mathrm{PDA} & \subseteq & \mathrm{BND\text{-}APDA} & \subseteq & \mathrm{ALT\text{-}PDA} \\
 & & \cup\!\vert & & \| \\
 & & \mathrm{NSPACE}(n) & & \\
\| & & \cup\!\vert & & \bigcup_{c>0} \mathrm{DSPACE}(c^n) \\
 & & \mathrm{DSPACE}(n) & & \| \\
 & & \cup\!\vert & & \\
\mathrm{CFL} & = & \mathrm{BND\text{-}ACFL} & \subsetneq & \mathrm{ALT\text{-}CFL}
\end{array}
$$

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における併列性ーとくにalternationについてー、L.A.ｼﾝﾎﾟｼﾞｳﾑ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