自然言語処理 63－1
（1987. 9. 25）

一般化文脈自由文法，多重文脈自由文法及び head grammarについて

嵩 忠雄　　関 浩之　　藤井 護
大阪大学基礎工学部情報工学科

Pollardが導入した一般化文脈自由文法の部分クラスとして，多重文脈自由文法（mcfg）などを導入し，それらの基本的性質を明らかにする．mcfgは，例えば，respectively文のような互いに割り込んだ構文要素間の依存関係を表現するのに適していて，生成能力が文脈自由文法（cfg）より大きく，文脈規定文法より小さい．mcfgではcfgにおける句構造，導出木などが自然な形で拡張され，cfgの主な特質が受けつがれる．次に，mcfgが生成する言語の所属問題は多項式オーダの時間的複雑度を持つことを示す．また，Pollardにより自然言語文法記述のために導入されたhead grammarは，多重文脈自由文法の特殊な部分クラスに相当することを示し，それが生成する言語の所属問題がn^6のオーダの複雑度であることを示す．これは，Pollardの結果に比べて1次の改善である．

Generalized Context-Free Grammars, Multiple Context-Free Grammars and Head Grammars

Tadao KASAMI　　Hiroyuki SEKI　　Mamoru FUJII

Department of Information and Computer Sciences
Faculty of Engineering Science, Osaka University
Toyonaka, Osaka, 560 JAPAN

We introduce multiple context-free grammars (mcfg's) as a subclass of generalized context-free grammars introduced by Pollard. mcfg's inherit the main properties of context-free grammars (cfg's) and the generative capacity of mcfg's is greater than that of cfg's and smaller than that of context-sensitive grammars. It is shown that the time complexity of the membership problem for the class of multiple context-free languages is polynomial. It is also shown that the class of head languages introduced by Pollard to describe natural language grammars is a special subclass of multiple context-free languages, and the time complexity of the membership problem for the class of head languages is of order n^6, which is an improvement over the order n^7 given by Pollard.

1．まえがき

文脈自由文法[1]（以下ｃｆｇと略記）が，自然語における，例えば，respectively文のような互いに割り込んだ構文要素間の依存関係（…A…B…A′…B′…）を表現するのに適していないと考えている言語学者が少なくない[2]。一方，文脈規定文法（Ｉ型文法[1]，以下ｃｓｇと略記）では生成能力が大きすぎるので，ｃｆｇとｃｓｇの中間のさまざまな形式文法が1960年代から1970年前半にかけて提案された[4][5]。当時の主たる関心は，文の生成，導出過程であって，句構造の適切な定義，構文解析の簡明な定式化が得られるような文法の提案ではなかった。もともと，ChomskyのＩ型，０型文法は生成文法ではあるが，ｃｆｇにおける句構造の"自然"な拡張であるような"句構造"は導入されておらず"句構造文法"と呼ぶのがふさわしいか疑問である。この意味で，Pollardが1984年に文献(6)で導入した一般化文脈自由文法（generalized context-free grammar，以下，ｇｃｆｇと略記，2．1参照）は興味ある定式化である。Pollard自身はその生成能力にふれていないが，2．2で述べるように，０型文法と生成能力が同一である。従って，一般形では能力が大きすぎる。Pullum-Gazdarらの主張[2]に従って，少なくとも構文解析が文の長さに対して多項式時間のオーダであるｇｃｆｇの部分クラスを考えるべきであろう。また，2．2で説明するように，一般のｇｃｆｇの句構造は抽象的なものであって，ｃｆｇの句構造の"自然な"拡張とは必ずしもいえない。

Pollardはｇｃｆｇの部分クラスとして，生成能力においてｃｆｇより大きいhead grammar（以下，ｈｇと略記）を導入し，ｈｇで生成される言語ｈｌのクラスの所属問題の時間的複雑度が文の長さｎに対して$O(n^7)$以下であることを示した。

ｈｇは自然語文法記述のために導入されたという意味で興味深いが，形式文法としてはかなり特殊な文法であるとの印象を与える。文法のクラスとしてｃｆｇの"自然な"拡張には必ずしもなっていない。例えば，文脈自由言語（以下ｃｆｌと略記）は代入によって閉じており[1]，その証明も簡単で，定義からほとんど自明である。一方ｈｌについて同様なことが成立すると予想されているが[6]証明は知られていない。

3．で，ｃｆｇ，ｈｇを含み，ｇｃｆｇの部分クラスである多重文脈自由文法（以下，ｍｃｆｇと略称）を導入する。ｍｃｆｇではｃｆｇにおける句構造，導出木などが自然な形で拡張される。3．ではさらに，(イ)ｍｃｆｇの生成能力がｃｆｇよりも大きくｃｓｇよりも小さいこと，(ロ)ｍｃｆｇでは，代入等の言語演算に関する閉包性，pumping lemma等ｃｆｇの主な特質が受けつがれることを述べる。

4．では，ｈｇがｍｃｆｇの特殊な部分クラスに相当することを示す。ｍｃｆｇ，ｐｍｃｆｇ，ｈｇが生成する言語のクラスをそれぞれL_{MCF}，L_{PMCF}，L_Hと書く。ついで，5．では，L_{MCF}，L_{PMCF}の所属問題の時間複雑度が文の長さｎの多項式オーダであることを示す。その系として，L_Hについては，n^6のオーダであることを示す。L_Hについては，Pollardがn^7のオーダであることを示している[6]ので，１次の改善である。

2．一般化文脈自由文法

2．1 一般化文脈自由文法の定義

一般化文脈自由文法（以下，ｇｃｆｇと略記）は次のような５字組（Ｎ，Ｏ，Ｆ，Ｐ，Ｓ）として定義される（文献(6)のAppendix 2，但し，文脈自由文法との対応が分かりやすいように，若干変更している）。

（Ｇ１）　Ｎは非終端記号の有限集合．

（Ｇ２）　Ｏは有限な記号集合の上の系列からなるある集合．

（Ｇ３）　ＦはＯの有限次元の直積集合からＯへの部分写像からなる有限集合。Ｆに属するO^mからＯへの部分写像全体の集合をF_mと書く．

（Ｇ４）　Ｐは$\bigcup_m (F_m \times N^{m+1})$の有限部分集合である．Ｐの元は（書換）規則と呼ばれ，$(f, C_0, C_1, \cdots, C_m)$と書く代りに，

$$C_0 \to f\,[C_1, C_2, \cdots, C_m]$$

と書く（$C_1, C_2, \cdots, C_m$は重複を許す）．とくに，$m=0$のとき，すなわちｆがＯの一つの元に等しいとき，その規則を終端規則と呼ぶ．

（Ｇ５）　$S \in N$．Ｓは始端記号と呼ばれる。□

ｇｃｆｇでは，以下に述べるようにｃｆｇにおける導出木を拡張した概念が定義されており，構文に基づいた変換，とくに文の形式的意味定義[6]などの枠組みとして適している。

$C \in N$について，$L_G(C)$を次の条件（Ｌ１）と（Ｌ２）を満たす最小集合として定義する。

（Ｌ１）　終端規則$C \to \alpha$があれば，$\alpha \in L_G(C)$。

（Ｌ２）　$\alpha_i \in L_G(C_i)$ $(1 \leq i \leq m)$，

$C \to f\,[C_1, C_2, \cdots, C_m] \in P$，かつ

$f(\alpha_1, \alpha_2, \cdots, \alpha_m)$が定義されているとき，

$$f(\alpha_1, \alpha_2, \cdots, \alpha_m) \in L_G(C).$$ □

$\alpha \in L_G(C)$であるとき，"αはＣに構文解析される"という。また，上の（Ｌ２）で仮定が成り立っているとき，α_i $(1 \leq i \leq m)$は$f(\alpha_1, \alpha_2, \cdots, \alpha_m)$の部分句であるという。βがαの部分句であり，γがβの部分句であるとき，γもαの部分句であるという。とくに，

$$L(G) \triangleq L_G(S)$$

と定義し，Ｌ(Ｇ)をＧによって生成される一般化文脈自由言語（略して，ｇｃｆｌと書く）という。

Ｇにおける導出木を次のように定義する。

（Ｔ１）　終端規則$C \to \alpha$に対して，根の節点（ラベルＣ）がただ一つの子節点（ラベルα）をもつ木は

αの導出木である。

（T２）　α_iの導出木をT_i ($1\leq i\leq m$)，その根のラベルをC_iとし，$C\rightarrow f\ [C_1, C_2, \cdots, C_m]\in P$，かつ$f(\alpha_1, \alpha_2, \cdots, \alpha_m)$が定義されているとき，根のラベルはC（必要なら，Cの代りに適用規則名），根から出る枝数がm，i番目の枝の端点（ラベルC_i）以下にT_iを部分木としてもつ木は$f(\alpha_1, \alpha_2, \cdots, \alpha_m)$の導出木である。

（T３）　それ以外に導出木はない。 □

αの導出木において，端点のラベルを連接して得られる系列が，必ずしもαに等しくないところが，ｃｆｇにおける導出木と異なる。

ｇｃｆｇ，ｇｃｆｌは，次の定理１が成り立つという意味で，それぞれ，ｃｆｇ，ｃｆｌの一般化になっている。

定理１ (Pollard[6])：与えられたｃｆｇ　G_0に対し，次の条件を満たすｇｃｆｇ　Gを構成できる。
条件「Gにおける$\alpha\in O$の導出木はG_0におけるαの導出木と等しく，$C\in N$，$\alpha\in T^*$について，
$\alpha\in L_G(C)$　<=>　$C \overset{*}{\underset{G_0}{\Rightarrow}} \alpha$。従って，$L(G)=L(G_0)$。」 □

２．２　一般化文脈自由文法の生成能力

ｇｃｆｇの定義の（G３）において，Fの関数として，部分的に計算可能な任意の関数を選ぶことを許すならば，すべての帰納的に可算な系列集合（０型文法で生成される言語[1]）がｇｃｆｌであることは容易に分かる。Fの関数をごく単純な基本関数の複合関数に限定しても同じ結論が得られる（部分的に可算な関数が，チューリング機械を用いて定義されることから予想がつくであろう）。すなわち，次の定理が成り立つ。

定理２：帰納的に可算な系列集合（０型文法で生成される言語）は，一般化文脈自由言語である。とくに，決定性チューリング機械Mを模擬して，Mが受理する言語を生成するｇｃｆｇを構成することができる。□

もちろん，この定理の逆も成立する。すなわち，一般化文脈自由文法G＝（O，N，F，P，S）において，Oが有限個の記号集合の上の系列全体の集合で，Fに属する関数が部分的に計算可能であるならば，Gが生成する言語は帰納的に可算な系列集合である。

ところで，ｇｃｆｇにおいて“自然な”句構造が定義されるためには，Fの関数について，（１）与えられた引数について，関数値が定義されているか否かを判別する述語が基本関数の複合関数として表現できること，及び，（２）関数の定義域において“情報損失”がないこと，すなわち，関数値（系列）から各引数の値（系列）を容易に復元できることが必要であろう。３．ではFに属する関数を，系列間の連接演算のみを用いて定義される関数に限定したｇｃｆｇの部分クラスについて述べる。

３．多重文脈自由文法

３．１　多重文脈自由文法の定義

正整数mに対して，次の条件（M１）～（M４）を満たすｇｃｆｇ　G＝（N，O，F，P，S）をm－並列多重文脈自由文法（以下，m－ｐｍｃｆｇと略記，mを明示しないときは，ｐｍｃｆｇ）と呼ぶ。TをNと互いに素である有限な終端記号集合とする。

（M１）　$O=\bigcup_{i=1}^{m}(T^*)^i$

（M２）　$f\in F$の引数の数を$a(f)$とする。各fについて，m以下の正整数$r(f)$，$d_i(f)$ ($1\leq i\leq a(f)$) が決まっていて，fは下記の条件（ｆ１）を満たす$(T^*)^{d_1(f)}\times(T^*)^{d_2(f)}\times\cdots\times(T^*)^{d_{a(f)}(f)}$から$(T^*)^{r(f)}$への関数である。$f^h$ ($1\leq h\leq r(f)$) をfの第h成分を表す関数とする。x_{ij} ($1\leq i\leq a(f)$, $1\leq j\leq d_i(f)$) をT^*の上の変数とし，

$\bar{x}_i \triangleq (x_{i1}, x_{i2}, \cdots, x_{id_i(f)})$
$X\triangleq\{x_{ij}\mid 1\leq i\leq a(f), 1\leq j\leq d_i(f)\}$

とおく。

（ｆ１）　$f^h(\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_{a(f)})$ ($1\leq h\leq r(f)$) はT^*の定系列とXに属するいくつかの変数の連接で表される。すなわち，

$f^h(\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_{a(f)}) \triangleq$
$\alpha_{h0}\, z_{h1}\, \alpha_{h1}\, z_{h2} \cdots z_{hv_h(f)}\, \alpha_{hv_h(f)}$　（＊１）

ここで，$\alpha_{hk}\in T^*$ ($0\leq k\leq v_h(f)$)，$z_{hk}\in X$ ($1\leq k\leq v_h(f)$)。

（M３）　各非終端記号$C\in N$について，正整数$d(C)$が定まっていて，Pが規則

$C\rightarrow f\ [C_1, C_2, \cdots, C_{a(f)}]$

を含めば，$r(f)=d(C)$，$d_i(f)=d(C_i)$ ($1\leq i\leq a(f)$) が成り立つ。従って，$\alpha\in L_G(C)$ならば，$\alpha\in(T^*)^{d(C)}$である。

（M４）　始端記号Sについて，$d(S)=1$。従って，$L(G)\subseteq T^*$。 □

Fの関数がさらに次の条件（ｆ２）も満たすとき，Gをm－多重文脈自由文法（m－ｍｃｆｇと略記，mを明記しないときは，ｍｃｆｇ）と呼ぶ。

（ｆ２）　Xの各変数x_{ij}に対して，$h=1$から$r(f)$までにわたって，（＊１）の右辺にx_{ij}が現れる総回数は１以下である。 □

（＊１）の右辺に同じ変数が二度以上現れたり，異なるhについて同一変数がまたがって現れていると，その変数に代入される系列が二度以上コピーされることになる。条件（ｆ２）は，このようなコピー操作をfの定義に利用するのを禁止するための条件である。

m－ｐｍｃｆｇ，m－ｍｃｆｇで生成される言語をそれぞれ，m－並列多重文脈自由言語，m－多重文脈自由言語（略して，m－ｐｍｃｆｌ，m－ｍｃｆｌと書く。mを指定しないときは，ｐｍｃｆｌ，ｍｃｆｌ）と呼ぶ。m－ｐｍｃｆｌ，ｐｍｃｆｌ，m－ｍｃｆｌ，ｍｃｆｌ，ｃｆｌ全体の集合を，それぞれ$L_{m\text{-}PMCF}$，L_{PMCF}，$L_{m\text{-}MCF}$，L_{MCF}，L_{CF}と書く。２．１で述べ

たこと，および，ｍｃｆｇの定義から，

$$L_{CF} = L_{1\text{-}MCF} \qquad (*2)$$

例１：

（１）　$\{\alpha^2 \mid \alpha \in \{0, 1\}^+\} \in L_{1\text{-}PMCF} \cap L_{2\text{-}MCF}$,

（２）　$\{a^{2^n} \mid n \geq 0\} \in L_{1\text{-}PMCF}$,

（３）　$\{a_1^n a_2^n \cdots a_{2m}^n \mid n \geq 0\} \in L_{m\text{-}MCF}$, $m \geq 1$,

（４）　$\{a^{n^2} \mid n > 0\} \in L_{2\text{-}PMCF}$.

（略証）（１）$L_{1\text{-}PMCF}$であること：$T \triangleq \{0, 1\}$，$N \triangleq \{C, S\}$，$P \triangleq \{C \to 0 \mid 1 \mid g_1[C, C]$，$S \to g_2[C]\}$，ここで，$g_1(x_1, x_2) \triangleq x_1 x_2$, $g_2(x) \triangleq xx$。（１′）$L_{2\text{-}MCF}$であること：Ｔ，Ｎは（１）と同じ．$P \triangleq \{C \to (0, 0) \mid (1, 1) \mid g_{3,0}[C] \mid g_{3,1}[C]$，$S \to g_4[C]\}$，ここで，$g_{3,a}((x_1, x_2)) \triangleq (a x_1, a x_2)$，$a=0, 1$, $g_4((x_1, x_2)) \triangleq x_1 x_2$.

（２）$N \triangleq \{S\}$，$P \triangleq \{S \to a, S \to g_2[S]\}$．

（３）$T \triangleq \{a_i \mid 1 \leq i \leq 2m\}$，$N \triangleq \{C, S\}$，$P \triangleq \{C \to (\varepsilon, \varepsilon, \cdots, \varepsilon) \mid g_5[C]$，$S \to g_6[C]\}$，ここで，$g_5((x_1, x_2, \cdots, x_m)) \triangleq (a_1 x_1 a_2, a_3 x_2 a_4, \cdots, a_{2m-1} x_m a_{2m})$，$g_6((x_1, x_2, \cdots, x_m)) \triangleq x_1 x_2 \cdots x_m$.

（４）$N \triangleq \{C, S\}$, $P \triangleq \{C \to (\varepsilon, \varepsilon) \mid g_7[C]$, $S \to g_8[C]\}$, ここで，$g_7((x_1, x_2)) \triangleq (a x_1, a x_1^2 x_2)$, $g_8((x_1, x_2)) \triangleq a x_1^2 x_2$. □

例１のどの系列集合もL_{CF}に属さない[4][5]。従って，

$$L_{CF} \subsetneq L_{1\text{-}PMCF} \cap L_{2\text{-}MCF} \qquad (*3)$$

Ｇをｍｃｆｇ，ｔを$\overline{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_k) \in L_G(C)$，ＣはＧの非終端記号，の導出木とする．$v_1$, v_2をｔの異なる内部節点とし，そのラベルをそれぞれC_1，C_2とする．v_1, v_2を根とするｔの部分木が導出する$L_G(C_1)$，$L_G(C_2)$の元をそれぞれ$\overline{\beta} = (\beta_1, \beta_2, \cdots, \beta_{k_1})$，$\overline{\gamma} = (\gamma_1, \gamma_2, \cdots, \gamma_{k_2})$とする．各$\beta_i$は$\overline{\alpha}$のたかだか一つの（Ｇが条件（ｆ３）を満たせば，ちょうど一つの）成分の部分列として含まれ，β_i, β_j $(i \neq j)$ が$\overline{\alpha}$の同じ成分の部分列として含まれても重なることはない．v_1とv_2との間に祖先－子孫の関係がなければ，β_iとγ_jが仮に$\overline{\alpha}$の同じ成分の部分列として含まれても互いに重なり合わない．このように，ｍｃｆｇでは，各非終端記号Ｃに対し$L_G(C)$の元は$d(C)$個の成分からなっているが，構文の中では，まとまったふるまいをし，“連句”と呼ぶにふさわしく，連句間に祖先－子孫関係が成り立つ．このことは，scattered context-free grammar[4], parallel context-free grammar[7][8]とか matrix grammar[4]（それぞれ，ｓｃｆｇ，ｐｃｆｇ，ｍｇと略記）と異なる所である．ｓｃｆｇでは，一つの規則は

$$(C_1 \to u_1, C_2 \to u_2, \cdots, C_\ell \to u_\ell)$$

但し，$C_i \in N$, $u_i \in (T \cup N)^*$, Ｎ，Ｔはそれぞれ，非終端記号及び終端記号の集合

の形であり，ｍｃｆｇの規則に一見似ている．$\alpha \in (T \cup N)^*$にC_1，C_2，…，C_ℓが現れているとき，それらにそれぞれu_1, u_2, …, u_ℓを代入して得られる系列βをαからこの規則で導出される系列と定義する．しかし，βにおけるこれらの部分列u_1, u_2, …, u_ℓはそれ以降の導出において何らまとまった働きをすることはない．連句としての祖先－子孫関係の有意義な定義は不可能であろう．ｐｃｆｇ，ｍｇについても同様である．ｐｃｆｇはコピー操作を伴う点で，ｐｍｃｆｇに少し似ている．しかし，ｐｃｆｇの生成する言語のクラスとｃｆｌは互いに他を含まない[8]点でも，ｐｍｃｆｇと異なる（（＊２），（＊３）参照）．

３．２　多重文脈自由言語の性質

ｐｍｃｆｌおよびｍｃｆｌの主な性質を述べる．

補題３：ｐｍｃｆｇ　Ｇが与えられたとき，$L(G)$がεを含むか否かは決定可能である． □

補題４：与えられたm－ｐｍｃｆｇ（または，m－ｍｃｆｇ）と弱等価で，次の情報無損失条件（ｆ３）及び条件（Ｎ１）と（Ｎ２）を満たすm－ｐｍｃｆｇ（または，m－ｍｃｆｇ）を構成できる．

（ｆ３）　Ｘのどの変数も，少なくとも一つのｈについて，（＊１）の右辺に少なくとも一回現れる．

（Ｎ１）　終端規則の左辺に現れるすべての非終端記号Ｃについて，$d(C) = 1$.

（Ｎ２）　始端記号以外のすべての非終端記号Ｃについて，$(\alpha_1, \alpha_2, \cdots, \alpha_{d(C)}) \in L_G(C)$ならば，$\alpha_i \neq \varepsilon$ $(1 \leq i \leq d(C))$． □

ｃｆｌに関するpumping lemma[1][4][5]に類似した次の補題が成り立つ．

補題５（pumping lemma）：m－ｍｃｆｌ　Ｌについて，Ｌが無限集合ならば，次の条件を満たす$u_j \in T^*$ $(1 \leq j \leq m+1)$，v_j, w_j, $s_j \in T^*$ $(1 \leq j \leq m)$ が存在する．

（１）　$\sum_{j=1}^{m} |v_j s_j| > 0$

（２）任意の非負整数ｉに対して，

$$z_i = u_1 v_1^i w_1 s_1^i u_2 v_2^i w_2 s_2^i u_3 \cdots u_m v_m^i w_m s_m^i u_{m+1} \in L. \;\square$$

補題６：$\{a_1^n a_2^n \cdots a_{2m+1}^n \mid n \geq 1\}$は$m$－ｍｃｆｌではない．

（略証）補題５より容易． □

$\{a_1^n a_2^n \cdots a_{2m+1}^n \mid n \geq 1\} \in L_{(m+1)\text{-}MCF}$（例１の（３）と同様にして示せる）と，補題６から，

定理７：$L_{m\text{-}MCF} \subsetneq L_{(m+1)\text{-}MCF}$ □

補題８：$\{a^{2^n} \mid n \geq 1\}$はどの$m$についても$L_{m\text{-}MCF}$に属さない．

（略証）補題５より容易． □

例１の（２）と補題８から，

定理９：$L_{1\text{-}PMCF} \not\subseteq L_{MCF}$ □

ｃｆｌ同様，次の定理が成り立つ．

定理１０：（１）クラス$L_{m\text{-}PMCF}$（または$L_{m\text{-}MCF}$）

は代入のもとで閉じている。

（2） $L_{m\text{-}PMCF}$（または $L_{m\text{-}MCF}$）は合併，連接，＊，＋に関して閉じている。

（3） $L \in L_{m\text{-}PMCF}$（または $L_{m\text{-}MCF}$），$R \in$ 正規言語とすると，$L \cap R \in L_{m\text{-}PMCF}$（または $L_{m\text{-}MCF}$）．□

代入で閉じているので準同型写像でも閉じている。

また，判定問題について，cflの場合と同様の論法で次の定理が証明される．

定理11：与えられたpmcfg Gに対し，L(G)が空か否かは決定可能である．□

共通集合が空か否かが決定不能であるような2つのcflの存在が知られているので[4][5]，この定理の系として，L_{PMCF}及びL_{MCF}は，共通集合をとる演算で閉じていないことが分かる．一方，併合について閉じているので，補集合に関しても，閉じていない．

さらに，次の定理が成り立つ．

定理12：L_{PMCF}は文脈規定言語のクラスL_{CS}に真に含まれる．

（略証）与えられたpmcfg Gに対して，L(G)を受理集合とする非決定性線形拘束オートマトンを構成できる．真に含まれることは，L_{CS}について空問題が決定不能であることと定理11による．□

4．head grammar

Pollardが導入したhead grammar[3][6]（以下hgと略記）は，特殊なgcfgである．hgでは，

$$O \triangleq \{(\alpha, i) \mid \alpha \in T^*, 1 \leq i \leq |\alpha|\} \cup \{(\varepsilon, 0)\}$$

である．$(\alpha, i) \in O$について，αの第i番目の記号を“head”と呼ぶ．headは，例えば，英語の“文における動詞”とか，“名詞句における名詞”のように，構文解析上“鍵”となる中心的役割を果す単語を指す．

利用される関数は次の2型5種類である．$\alpha \in T^*$のi番目からj番目までの記号からなる部分列を$\alpha(i, j)$と書く．但し，$1 \leq i \leq j \leq |\alpha|$を満たさないi，jについては，$\alpha(i, j) = \varepsilon$と規約する。

（1） 連接型

正整数h，s（$h \leq s$）とT^*の定系列$\gamma_0, \gamma_1, \cdots, \gamma_s$が定まっていて，$(\alpha_j, i_j) \in O$（$1 \leq j \leq s$）に対して，

$$c((\alpha_1, i_1), (\alpha_2, i_2), \cdots, (\alpha_s, i_s)) \triangleq (\gamma_0 \alpha_1 \gamma_1 \alpha_2 \cdots \gamma_{s-1} \alpha_s \gamma_s, \sum_{i=0}^{h-1} |\gamma_i| + \sum_{i=1}^{h-1} |\alpha_i| + i_h)$$

但し，引数がすべて$(\varepsilon, 0)$であるとき，これらの関数値は$(\varepsilon, 0)$と定義される．また，$(\alpha_h, i_h) = (\varepsilon, 0)$かつ，あるp（$p \neq h$，$1 \leq p \leq s$）があって$(\alpha_p, i_p) \neq (\varepsilon, 0)$のとき，関数値は定義されない．

（2） 包み込み型

$(\alpha, i), (\beta, j) \in O$とする．$\alpha$のheadの直前か直後で$\alpha$を二分し，その間に$\beta$をはさみ込み，headとして$\alpha$か$\beta$のheadを使う．4通りの組み合わせがある．

$$w_1((\alpha, i), (\beta, j)) \triangleq (\alpha(1, i)\beta\alpha(i+1, |\alpha|), i)$$

$$w_2((\alpha, i), (\beta, j)) \triangleq (\alpha(1, i)\beta\alpha(i+1, |\alpha|), i+j)$$

$$w_3((\alpha, i), (\beta, j)) \triangleq (\alpha(1, i-1)\beta\alpha(i, |\alpha|), |\beta|+i)$$

$$w_4((\alpha, i), (\beta, j)) \triangleq (\alpha(1, i-1)\beta\alpha(i, |\alpha|), i+j-1)$$

但し，引数がともに$(\varepsilon, 0)$であるとき，これらの関数値は$(\varepsilon, 0)$と定義される．また，w_1，w_3について，$(\alpha, i) = (\varepsilon, 0)$，$(\beta, j) \neq (\varepsilon, 0)$のとき，$w_2$，$w_4$については，$(\alpha, i) \neq (\varepsilon, 0)$，$(\beta, j) = (\varepsilon, 0)$のとき，関数値は定義されない．

L(G)に対して，

$$\{\alpha \mid (\alpha, i) \in L(G)\}$$

をhg Gで生成される基底言語と呼ぶ．hgの文法によって生成される基底言語をhlとよび，hlのクラスをL_Hと書く．

例2：Gを，次のような規則をもつhgとする：$S \rightarrow (aa, 2) \mid (bb, 2) \mid c_a(D) \mid c_b(E)$，$D \rightarrow w_3(S, A)$，$E \rightarrow w_3(S, B)$，$A \rightarrow (a, 1)$，$B \rightarrow (b, 1)$，ただし，$x = a$，bに対し，$c_x((\alpha, i)) = (\alpha x, i)$．このとき，Gで生成される基底言語は$\{\alpha^2 \mid \alpha \in \{a, b\}^+\}$．□

次の条件を満たすhgを標準形と呼ぶ．

（1） 終端規則の右辺は，$(\varepsilon, 0)$か$(a, 1)$（$a \in T$）である．

（2） 連接型の関数は，2引数（$s = 2$）かつ定系列はすべてεである．（従って，これらの関数値はhの値（1または2）のみによって定まる．これをc_hと表す．）

（3） $C \rightarrow (\varepsilon, 0) \in P$なら，Cはどの規則の右辺にも現れない．

cfgのチョムスキー標準形の場合と同様の議論で，与えられたhgと弱等価な標準形を構成できる[6]．

次に与えられた標準形hg $G = (N, O, F, P, S)$に対して，Gで生成される基底言語Lを生成する2−mcfg $G' = (N', O', F', P', S')$の作り方を示す．

1） S'をNに含まれない記号として，

$$N' \triangleq \{C\text{-}a \mid C \in N, a \in T\} \cup \{S'\}.$$

2） $O' \triangleq T^* \cup (T^*)^2$.

3） $(\varepsilon, 0)$をεで表す．それ以外のOの元(α, i)については，次の対応3.1）と3.2）がつくように，F'，P'を定義する．

3.1） $(\alpha, i) \in L_G(C)$
$\Rightarrow (\alpha(1, i-1), \alpha(i+1, |\alpha|)) \in L_{G'}(C\text{-}\alpha(i, i))$

3.2） $(\alpha_1, \alpha_2) \in L_{G'}(C\text{-}a)$
$\Rightarrow (\alpha_1 a \alpha_2, |\alpha_1|+1) \in L_G(C)$

c_h (h＝1, 2), w_ℓ $(1 \leq \ell \leq 4)$ に対応して，各 $a \in T$ に対して，次のような $(T^*)^2 \times (T^*)^2$ から $(T^*)^2$ への関数 c_h-a, w_ℓ-aを導入する。

$c_1\text{-a}((x_1, x_2), (y_1, y_2)) \triangleq (x_1, x_2 y_1 a y_2)$,

$c_2\text{-a}((x_1, x_2), (y_1, y_2)) \triangleq (x_1 a x_2 y_1, y_2)$,

$w_1\text{-a}((x_1, x_2), (y_1, y_2)) \triangleq (x_1, y_1 a y_2 x_2)$,

$w_2\text{-a}((x_1, x_2), (y_1, y_2)) \triangleq (x_1 a y_1, y_2 x_2)$,

$w_3\text{-a}((x_1, x_2), (y_1, y_2)) \triangleq (x_1 y_1 a y_2, x_2)$,

$w_4\text{-a}((x_1, x_2), (y_1, y_2)) \triangleq (x_1 y_1, y_2 a x_2)$。

4) Pに対応して，P′を次の条件を満たす最小集合とする。

P′1) $S \to (\varepsilon, 0) \in P$ なら，

$S' \to \varepsilon \in P'$

P′2) $C \to (a, 1) \in P$ なら，

$C\text{-a} \to (\varepsilon, \varepsilon) \in P'$

P′3) $C \to c_h [C_1, C_2] \in P$ (h＝1, 2) ならば，すべてのa，b∈Tについて，

P′3.1) h＝1のとき，

$C\text{-a} \to c_1\text{-b} [C_1\text{-a}, C_2\text{-b}] \in P'$

P′3.2) h＝2のとき，

$C\text{-a} \to c_2\text{-b} [C_1\text{-b}, C_2\text{-a}] \in P'$

P′4) $C \to w_\ell [C_1, C_2] \in P$ $(1 \leq \ell \leq 4)$ ならば，すべてのa，b∈Tについて，

P′4.1) ℓ＝1または3のとき，

$C\text{-a} \to w_\ell\text{-b} [C_1\text{-a}, C_2\text{-b}] \in P'$

P′4.2) ℓ＝2または4のとき，

$C\text{-a} \to w_\ell\text{-b} [C_1\text{-b}, C_2\text{-a}] \in P'$

P′5) すべての $a \in T$ について，

$S' \to \text{ins-a} [S\text{-a}]$,

ここで，ins-aは次のように定義される。

$\text{ins-a}((x_1, x_2)) \triangleq x_1 \ a \ x_2$

5) S′は始端記号。 □

上で定義したG′は，条件（f3）を満たし，かつ規則の右辺に表れる非終端記号の数は高々2である。補題4より，G′と弱等価でさらに条件（N1）と（N2）を満たす2－mcfg G″を構成できる。また詳細は省略するが，とくに，規則の右辺に表れる非終端記号が高々2であるように，G″を構成できる。以上まとめて，

定理13：hg Gが与えられたとき，Gが生成する基底言語を生成し，条件（f3），（N1），（N2）を満たし，規則の右辺に現れる非終端記号の数が高々2であるような2－mcfg G′を構成することができる。 □

なお，hg Gに対し，(β, i) が $(\alpha, j) \in L(G)$ の部分句であるとすると，$\beta(1, i-1)$，$\beta(i+1, |\beta|)$ は α に部分系列として含まれている。headの $\beta(i, i)$ は α において，それらの部分系列の間に現われているが，その位置はそのいずれかに隣接しているとは限らない。例えば，$A \to (a_1 a_2 a_3, 2)$，$B \to (b, 1)$，$C \to w_1(A, B)$，$S \to w_3(C, B)$ とすれば，$\beta = (a_1 a_2 a_3, 2)$ は $\alpha = (a_1 b a_2 b a_3, 3)$ の部分句であるが，β のhead a_2 は α において a_1，a_3 のいずれにも隣接していない。従って，(β, i) と表現するより，系列の対 $((\beta(1, i-1), \beta(i+1, |\beta|))$ と記号 $\beta(i, i)$ の組み合わせで現す方がより実態を反映している。

5．多重文脈自由言語の所属問題

Gを補題1の条件（f3），（N1）と（N2）を満たすmcfgとする。与えられた $\alpha \in T^*$ に対して，$\alpha \in L(G)$ か否かを判定する問題を考える。α の互いに素な部分列 $\alpha(\ell_1, r_1)$，$\alpha(\ell_2, r_2)$，…，$\alpha(\ell_k, r_k)$（順番は α 上の順と一致する必要はない）が $C \in N$ に構文解析されるとき，すなわち，$(\alpha(\ell_1, r_1), \alpha(\ell_2, r_2), \cdots, \alpha(\ell_k, r_k)) \in L_G(C)$ であるとき，この部分系列の組を単にC－連句といい，$\alpha(\ell_i, r_i)$ をその第i分句と呼ぶ。また $(\ell_1, r_1, \ell_2, r_2, \cdots, \ell_k, r_k)$ をこの連句の位置ベクトルと呼ぶことにする。位置ベクトルを与えれば部分列の系列が一意に定まるから，位置ベクトルで連句を表す。$r_1, r_2, \cdots, r_k$ の最大値を連句の右端位置という。

C－連句 α の“構文解析”（2.の（L1），（L2）より α を得る過程）で，最後の適用規則が

$r: C \to f [C_1, C_2, \cdots, C_{a(f)}]$

であるとする。$C_1, C_2, \cdots, C_{a(f)}$ の互いに素な連句（子部分連句）から，fの定義によって親のC－連句が作られる。これらの子部分連句を順に第1子連句，第2子連句，…と呼ぶ。また，子部分連句のうち右端位置が最大のもの（ちょうど一つある）を右端子連句と呼ぶ。第i子連句の位置ベクトルを $(\ell_1^{(i)}, r_1^{(i)}, \cdots, \ell_{d(C_i)}^{(i)}, r_{d(C_i)}^{(i)})$ と書く。

一般に，任意の $\alpha \in L(G)$ に対し，規則rを適用するときの子部分連句の可能な位置ベクトルの総数が高々 $O(n^e)$（$n=|\alpha|$，以下でも同様）であるとき，規則rの自由度D(r)がe以下であるという。Gのどの規則の自由度もe以下であるとき，Gの自由度D(G)がe以下であるという。次にD(r)の評価を考える。

子連句の位置ベクトルの成分の総数は，$2 \sum_{i=1}^{a(f)} d(C_i)$ である。一方，（＊1）式において，例えば，z_{h1} が x_{21}，z_{h2} が x_{13} とすると，系列 α 中において，第2子連句の第1分句の直後に定数系列 α_{h1} が続き，その直後に第1子連句の第3分句が続いていなければならない。従って，$\ell_3^{(1)} = r_1^{(2)} + |\alpha_{h1}| + 1$ の関係が成り立たねばならない。このような，式（＊1）によって指定される子連句の分句間の隣接関係による位置ベクトル成分間の制約式の個数は，f_h の定義式に現れている変数の個数 $v_h(f)$ から1引いたものに等しい。条件（f2），（f3）から，制約式の総数は，

$\{\sum_{i=1}^{a(f)} d(C_i)\} - d(C_0)$

に等しい。これらの制約式は，$\ell_j^{(i)}$，$r_j^{(i)}$ を変数とみ

たとき，1次式として1次独立であるから，子連句の位置ベクトル成分中独立でありうるものの総数は，

$$d(C_0)+\sum_{i=1}^{a(f)} d(C_i) \qquad (*4)$$

を越えない．従って，この値をdと書くと，$D(r)\leq d$である．さらに，例えば，カテゴリC_1の第2成分の長さがあるきまった長さ以下であることが分かっているとすると，$D(r)\leq d-1$となる．

以下，簡単のため各非終端規則について，（＊1）の右辺の定系列$\alpha_{h0}, \alpha_{h1}, \cdots, \alpha_{hv_h}$がすべてのhについて空系列であると仮定する．空系列でないとき，α_{hi}専用の非終端記号と終端規則を付け加えることにより，弱等価なmcfgが得られ，上で述べたように，新しい非終端記号を用いて書き直して得られる規則の自由度は変わらない．

$I \triangleq \{1, 2, \cdots, n, *\}$とおく．$C\in N$，$1\leq j\leq n$，$\bar{u}=(u_1, u_2, \cdots, u_{2d(C)})\in I^{2d(C)}$について，

$p(C, j, \bar{u})\triangleq$

$\{\bar{v}=(v_1, \cdots, v_{2d(C)})\mid \bar{v}$は右端位置がj以下のC－連句の位置ベクトルで，かつ，$v_i=u_i$または$u_i=*$ $(1\leq i\leq 2d(C))\}$

と定義する．$\bar{u}$を制約ベクトルという．$\bar{u}$の成分の値が＊であることは，その成分についての制約がないことを表している．$\bar{u}$のなかの成分中値が＊である成分の個数を$\nu(\bar{u})$と書くと，定義から，

$$|p(C, j, \bar{u})|\leq j^{\nu(\bar{u})}\leq n^{\nu(\bar{u})}$$

次に，cfℓについてのn^3認識アルゴリズムのmcfgへの一つの拡張を示す．

（a）　j＝1から順にnまで，手続きb(j)（（b）参照）を行う．b(n)が終ったとき，$p(S, n, (1, n))$が求まっており，

$$p(S, n, (1, n))\neq\phi \iff \alpha\in L(G)$$

である．

（b）　b(j)開始時には，各$C\in N$と$\bar{u}\in I^{2d(C)}$について，$p(C, j-1, \bar{u})$が表として求まっているとする．但し，$p(\cdot, 0, \cdot)$は空表，またj以上の値の成分をもつ$\bar{u}$に対し，$p(\cdot, j-1, \bar{u})$は空表である．手続きb(j)の目的は，$p(C, j, \bar{u})$を求めることである．各$C\in N$について，ワーキング用の表$\Delta p(C)$を用いる．$\Delta p(C)$の初期値は空表である．b(j)は，右端位置がjであるC－連句（の位置ベクトル，以下両者を一々区別しない）を$\Delta p(C)$に逐次格納すると共に，格納された連句について，それらを右端子連句とする親連句をすべて求める．まず，

（b1）　終端規則のみの適用によって構文解析される右端位置jの（連）句を非終端記号別に$\Delta p(C)$に登録する．次に，

（b2）　各$C\in N$について，$\Delta p(C)$に登録されている位置ベクトルのなかに完了マークのついていないものがあれば，その一つ$\bar{v}_1$を選び，完了マークをつけた後，手続き$c(C, \bar{v}_1, j)$（（c）参照）を行う．すべての$C\in N$について，$\Delta p(C)$がマークをもたないベクトルを含まなければ，右端位置がjのすべての連句が$\bigcup_{C\in N}\Delta p(C)$に求まっている．

$\Delta p(C)$の各連句を，それが満たす$2^{2d(C)}$個の制約条件$\bar{u}$について，表$p(C, j-1, \bar{u})$に付け加えて$p(C, j, \bar{u})$を求める．

（c）　規則

$$\phi:\ C_0\rightarrow f\ [C_1, C_2, \cdots, C_{a(f)}]$$

と，右端位置がjのC_{i_1}－連句$\bar{v}_1$，右端位置がjより小さいC_{i_2}－連句$\bar{v}_2$，…，C_{i_s}－連句$\bar{v}_s$（但し，$i_1, i_2, \cdots, i_s$は互いに異なるa(f)以下の正整数）が与えられたとき，

P1）$\bar{v}_1$を右端子連句とし，

P2）$\bar{v}_t$ $(1\leq t\leq s)$を第i_t－子連句としてもち，

P3）規則ϕを最後に適用して構文解析される，C_0－連句が存在するための必要条件（$s=a(f)$ならば十分条件でもある）は，

A1）$\bar{v}_1=(\ell_1, r_1, \cdots, \ell_k, r_k, \cdots)$，$r_k=j$とすると，変数$x_{i_1k}$があるhに関する（＊1）の右辺の右端に現れている．すなわち，

$$z_{hv_h}=x_{i_1k}.$$

A2）$\bar{v}_1, \cdots, \bar{v}_s$は互いに素（それらが表す連句が重なり合わない）であり，

A3）（＊1）によって指定される隣接関係に関する制約を$\bar{v}_1, \cdots, \bar{v}_s$の成分がすべて満たしている．□

手続き$c(C, \bar{v}_1, j)$は，次の条件Qを満たす各規則ϕ: $C_0\rightarrow f\ [C_1, C_2, \cdots, C_{a(f)}]$とa(f)以下の正整数$i_1$（以下，簡単のため$i_1=1$としている．必要なら引数の順番をかえた新しい関数を導入すればよい）について手続き$e(\phi, (\bar{v}_1))$（（e）参照）を実行する．

Q：「$C_{i_1}=C$，かつ条件A1）が成り立つ．」

手続き$e(\phi, (\bar{v}_1, \bar{v}_2, \cdots, \bar{v}_s))$（但し，$s<a(f)$，（e）参照）は，$i_1=1, i_2=2, \cdots, i_s=s$として，条件A1），A2），A3）が満たされているとき，$\bar{v}_1$を右端子連句，$\bar{v}_i$ $(1\leq i\leq s)$を第i－子連句としてもち，規則ϕを最後に適用して構文解析されるすべての連句を求め，それらのうち$\Delta p(C)$にまだ登録されていないものを付け加える．

（e）　手続き$e(\phi, (\bar{v}_1, \bar{v}_2, \cdots, \bar{v}_s))$では，まず第1～第s－子連句，$\bar{v}_1, \bar{v}_2, \cdots, \bar{v}_s$からきまる，第(s+1)－子連句の位置ベクトルに関するすべての制約を表す制約ベクトル$\bar{u}$を求める．表$p(C_{s+1}, j-1, \bar{u})$が空でないとき（空ならリターン），それに含まれる各位置ベクトル$\bar{v}_{s+1}$について，$\bar{v}_1, \bar{v}_2, \cdots, \bar{v}_{s+1}$が互いに素ならば（$\bar{v}_1, \cdots, \bar{v}_{s+1}$は$i_1=1, \cdots, i_{s+1}=s+1$として条件A1），A2），A3）を満たす），

（e1）　$s+1<a(f)$のとき，$e(\phi, (\bar{v}_1,$

$\bar{v}_2, \cdots, \bar{v}_{s+1})$)を実行する。

(e 2)　$s+1=a(f)$ならば，$\bar{v}_1$を右端子連句，$\bar{v}_i$ $(1 \leq i \leq s+1)$ を第i－子連句とするC_0－連句が求まったから，その位置ベクトルを$\Delta p(C_0)$が含んでいなければ$\Delta p(C_0)$に付け加える。　□

上述の手続きを実行する計算時間を評価するため次の仮定を置く。

(1)　表$p(C, j, \bar{u})$を，$\bar{u}$から一定の計算できまる先頭番地からのリストで実現する。どの位置ベクトルまで調べたかを示すポインタとリストの最後を示すポインタを使う。

(2)　位置ベクトル$\bar{v}$が与えられたとき，$\bar{v}$から一定の計算できまる番地のメモリの内容が1か0かによって，$\bar{v} \in \Delta p(C)$か否かを判定する。また$\Delta p(C)$自身は，ポインタ付リストで実現し，新しい位置ベクトルを逐次付け加える。マークが付いていない最初の位置ベクトルと，リストの最後を示す2つのポインタを用いる。

(3)　大きさn以下の整数演算が定数ステップでできる（漸近的にはlog nの多項式オーダ）。

Gの自由度D(G)がe以下であるとすると，各jについて，処理する子連句の候補の組み合わせの数はオーダ$O(n^{e-1})$（右端が指定されているから）であり，各組み合わせについて，行う基本操作回数は入力系列長に依存せず，定数でおさえられるので，全体でオーダ$O(n^e)$である。以上から，

定理14：Gを補題1の条件（f 3），条件（N 1）と（N 2）を満たすmcfgとし，その自由度D(G)がe以下であるとする。与えられた$\alpha \in T^*$に対して，$\alpha \in L(G)$か否かは$O(|\alpha|^e)$の計算時間† で判定できる。　□

定理13，14と（＊4）から，次の系が得られる。

系15：L_Hの所属問題の時間的複雑さは，オーダ$O(n^6)$である。　□

Pollardの結果に比べて1次の改善である。なお，Pollardは，hgで用いる規則にある制限をつけた場合にオーダがn^6に下がることを指摘しているが[6]，そのような制限によって，文法の生成能力が落ちるかどうかは，不明である。

なお，Gと弱等価なmcfg　G′でD(G′)ができるだけ小さいものを求めることが問題となる。cfgのとき，チョムスキー標準形G′ではD(G′)が3以下である。もちろん，$|\alpha|$に関するオーダのみを問題にする場合についての話である。

以上，簡単のため，mcflについて考えたが，上述の判定方法をpmcflに拡張し，多項式オーダ$O(n^{e+1})$の判定が可能であることを示すことができる。違う所は，式（＊1）において，（h＝1からqにわたって）二度以上現れる変数に対応する分句のコピーを考えねばならない所である。コピーになっているか否かの判定にnのオーダがかかるので，全体のオーダはn^{e+1}となる。なお，mcflの場合と比べて，（＊1）での変数の重複分だけ位置ベクトルの個数が多くなるが，一方，互いにコピーになっている分句の長さが等しいという制約条件が付け加わる。

† 厳密にはlog $|\alpha|$の多項式のオーダとの積となる。

6．むすび

gcfgの部分クラスとしてmcfgを導入し，mcfgの生成能力及びその諸性質について述べた。また，L_{MCF}の所属問題が多項式オーダの時間複雑度をもつこと，hgがmcfgの部分クラスに相当することを示し，それらの系として，L_Hの所属問題が$O(n^6)$の複雑度であることを示した。紙面の都合でいくつかの定理の証明を省略したが，それらについては文献(9)を参照されたい。hgに関する残された課題として，L_Hが代入について閉じているかどうか等がある。

謝辞　Pollardの文献等について御教示をいただいた，本学言語文化部の郡司隆男助教授に深謝します。また，種々御協力いただいた，本学基礎工学部の藤原　融助手及び，本学学生の松村　享，並河　英二の各氏に感謝します。

文　献

(1) 福村，稲垣：“オートマトン・形式言語理論と計算論”，岩波講座　情報科学6，岩波書店（昭57）。

(2) G. K. Pullum and G. Gazdar: “Natural languages and context-free languages”, Linguistics and Philosophy, 4, pp. 471-504 (1982).

(3) 郡司隆男：“句構造文法”，情報処理，27，8，pp. 868-875 (昭61-08).

(4) A. Salomma: “Formal Languages”, Academic Press, New York, U. S. A. (1973).

(5) J. Hopcroft and J. D. Ullman: “Introduction to Automata Theory, Languages and Computation, Addison-Wesley, Reading, Mass., U. S. A. (1979).

(6) C. J. Pollard: “Generalized phrase structure grammars, head grammars, and natural language”, Ph. D. dissertation, Stanford University (Feb. 1984).

(7) 鳥居，有沢：“同時導出による句構造言語”，信学論(C)，54-C，2，pp. 124-131 (昭46-02).

(8) S. Skyum: “Parallel context-free Languages”, Information and control, 26, 3, pp. 280-285 (Nov. 1974).

(9) 嵜，関，藤井：“一般化文脈自由文法と多重文脈自由文法”，信学論(D)（投稿中）。