2E−6

多重文脈自由文法のある部分クラスに対する効率の良い構文解析法について

中西　隆一+　　関　浩之+　　藤井　護++　　嵩　忠雄+

+大阪大学基礎工学部情報工学科　++大阪大学教養部

1．まえがき

自然言語の構文記述のための形式文法として，多重文脈自由文法（以下mcfgと略記）が提案されている(3)．mcfgでは，respectively文や倒置文のような互いに割り込んだ構文を自然に記述することができる．その生成能力は，文脈自由文法（以下cfgと略記）や，同じく自然言語の構文記述のためにPollardによって導入されたhead grammar(2)よりも真に大きく，文脈規定文法よりも真に小さい．また，mcfg Gに対して，入力系列αがGの生成する言語に属するかどうかを判定する時間計算量$O(n^e)$（nは入力系列長）のアルゴリズムが提案されている(4)．ここで，eはGのみに依存して定まる定数であるが，その上界は知られていない．

本稿では，mcfgのある部分クラスに対する時間計算量$O(n^2)$の構文解析アルゴリズムを提案する．この部分クラスに属するmcfgの生成する言語のクラスは，あいまいでないcfgの生成する言語をすべて含み，言語$\{a_1^p b_1^q a_2^p b_2^q \ldots a_m^p b_m^q \mid p,q \geq 1\}$ $(m \geq 1)$，$\{wcw \mid w \in \{0,1\}^+\}$等も含む．

2．多重文脈自由文法

正整数mに対して，m-多重文脈自由文法（m-mcfgと略記．mを明示しないときはmcfg）は次の[M1]～[M5]を満たす5字組$G=(N,O,F,P,S)$として定義される(3)．

[M1]Nは非終端記号の有限集合．各$A \in N$について，m以下の正整数$d(A)$が定まっているとする．

[M2]$O=\bigcup_{i=1}^{m}(T^*)^i$．TはNと互いに素である終端記号の有限集合．

[M3]$F=\quad\cup\quad F\{d_1,d_2,\ldots,d_q;r\}$．

$0<r\leq m;0\leq q;0<d_1,\ldots,d_q\leq m$

ここで，$F\{d_1,d_2,\ldots,d_q;r\}$は，次の条件[f1],[f2]をともに満たす$(T^*)^{d1}\times(T^*)^{d2}\times\cdots\times(T^*)^{dq}$から$(T^*)^r$への写像からなるある集合．$\mathbf{x}_i=(x_{i1},x_{i2},\cdots,x_{idi})$を$(T^*)^{di}$の上の変数とし，$X \triangleq \{x_{ij} \mid 1\leq i\leq q,\ 1\leq j\leq d_i\}$とおく．

[f1]fの第h成分f^h（$1\leq h\leq r$）の値は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，

$$f^h[\mathbf{x}_1,\mathbf{x}_2,\cdots,\mathbf{x}_q] \triangleq \alpha_{h0}\, z_{h1}\, \alpha_{h1}\, z_{h2} \cdots z_{hvh}\, \alpha_{hvh} \qquad (*1)$$

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

[f2]Xの各変数x_{ij}（$1\leq i\leq q,\ 1\leq j\leq d_i$）に対して，h＝1からq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．

[M4]Pは$A_0\rightarrow f[A_1,A_2,\cdots,A_q]$の形の規則の有限集合．ここで，$A_i\in N$ $(0\leq i\leq q)$，$f\in F\{d(A_1),d(A_2),\ldots,d(A_q);d(A_0)\}$．特に，q=0のときの規則を終端規則と呼び，それ以外の規則を非終端規則と呼ぶ．

[M5]$S\in N$．Sは始記号と呼ばれる．

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

[L1]Pに終端規則$A\rightarrow\alpha$があれば，$\alpha\in L_G(A)$．

[L2]$\alpha_i\in L_G(A_i)$ $(1\leq i\leq q)$ かつ$A\rightarrow f[A_1,A_2,\cdots,A_q]\in P$であるとき，$f[\alpha_1,\alpha_2,\cdots,\alpha_q]\in L_G(A)$．

[M4]から，$L_G(A)\subseteq(T^*)^{d(A)}$が成り立つ．$L_G(A)$の元をAから生成される連句という．$L(G)\triangleq L_G(S)$と定義し，L(G)をmcfg Gの生成する言語と呼ぶ．

［例1］$G=(\{S,A,B\},T\cup T^2,F,P,S)$，但し，$T=\{a,b,c,d\}$，$P=\{S\rightarrow f_1[A,B],\ A\rightarrow f_2[A]\mid(a,c),\ B\rightarrow f_3[B]\mid(b,d)\}$（$A\rightarrow\alpha\mid\beta$は$A\rightarrow\alpha$，$A\rightarrow\beta$を表す），$f_1[(x_1,x_2),(y_1,y_2)]\triangleq x_1y_1x_2y_2$，$f_2[(x,y)]\triangleq(ax,cy)$，$f_3[(x,y)]\triangleq(bx,dy)$とする．$L_G(A)=\{(a^p,c^p)\mid p\geq1\}$，$L_G(B)=\{(b^q,d^q)\mid q\geq1\}$，$L(G)=L_G(S)=\{a^pb^qc^pd^q\mid p,q\geq1\}$である．□

Gにおける導出木を次の[T1]～[T3]のように定義する．

[T1]終端規則$A\rightarrow\alpha$に対して，根の節点（ラベルA）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τ_i（$1\leq i\leq q$），その根のラベルをA_iとし，$A\rightarrow f[A_1,A_2,\cdots,A_q]\in P$とする．根のラベルがA（必要なら，Aの代りに適用規則名），根から出る枝数がq，i番目の枝の端点（ラベルA_i）以下にτ_iを部分木としてもつ木は$f[\alpha_1,\alpha_2,\cdots,\alpha_q]$の導出木である．

[T3]それ以外に導出木はない．

次の補題が成り立つ．

［補題1(3)］m-mcfg Gが与えられたとき，L(G)を生成し，次の情報無損失条件[f3](3)，および条件[N1]をともに満たすm-mcfgを構成できる．

[f3]Xのどの変数も，ちょうど一つのhについて，式(*1)の右辺にちょうど一回現れる．

[N1]各規則の右辺には，同じ非終端記号は2回以上現れない．□

以下では，上の条件[f3],[N1]をともに満たすようなmcfgのみを考える．

3．mcfgから派生するcfg

任意のmcfg $G=(N,O,F,P,S)$に対して，次の(1),(2)を満た

An Efficient Parsing Algorithm for a Subclass of Multiple Context-free Grammars

Ryuichi Nakanishi, Hiroyuki Seki, Mamoru Fujii and Tadao Kasami

Osaka University

すcfg $G'=(N',T,P',S^{[1]})$を，Gから派生するcfgという．

(1)$N'=\{A^{[h]}|A\in N,\ 1\leqq h\leqq d(A)\}$.

(2)規則$\rho:A\rightarrow f[A_1,\ldots,A_n]$右辺のf∈Fの定義式(*1)において，$z_{hk}=x_{p(hk)q(hk)}$ $(1\leqq h\leqq d(A),\ 1\leqq k\leqq v_h)$とする．

$\rho:A\rightarrow f[A_1,\ldots,A_n]\in P$ <=>

各h $(1\leqq h\leqq d(A))$について，$R_h:A^{[h]}\rightarrow$

$\alpha_{h0}A_{p(h1)}^{[q(h1)]}\ldots A_{p(hvh)}^{[q(hvh)]}\alpha_{hvh}\in P'$.

上で，G'の規則の組$(R_1,\ldots,R_{d(A)})$はGの規則ρから派生するという．

［例2］例1のmcfg Gから派生するcfgは，$G'=(\{S^{[1]},A^{[1]},A^{[2]},B^{[1]},B^{[2]}\},T,P',S^{[1]})$である．ここで，$P'=\{S^{[1]}\rightarrow A^{[1]}B^{[1]}A^{[2]}B^{[2]},\ A^{[1]}\rightarrow aA^{[1]}|a,\ A^{[2]}\rightarrow cA^{[2]}|c,\ B^{[1]}\rightarrow bB^{[1]}|b,\ B^{[2]}\rightarrow dB^{[2]}|d\}$である． □

cfg G_1に対して，G_1の生成する言語を$L(G_1)$と書く．Gを任意のmcfgとし，G'をGから派生するcfgとする．定義より，$L(G)\subseteq L(G')$であることが容易に示せる．

tを根のラベルが始記号であるようなG'における任意の導出木とする．tにおける連句頂点ベクトルおよび，tに対応するGにおける導出木プレフィクスを以下の[C1]～[C3]のように定義する．高さkの導出木プレフィクスは，G'の終端記号の組または非終端記号をラベルとしてもつ頂点からなる高さ（根から葉頂点までの道の最大長）kの木であり，非終端記号をラベルとしてもつ深さi$(1\leqq i<k)$の頂点には，tにおける深さiの連句頂点ベクトルが対応づけられる．

[C1]唯一つの頂点v_0からなる木は，tに対応するGにおける高さ0の導出木プレフィクスである．但し，tの根をrとするとき，v_0には連句頂点ベクトル(r)が対応づけられる．

[C2]τをtに対応する高さk-1の導出木プレフィクスであるとする．vをτにおけるラベルが終端記号でない深さk-1の任意の頂点とし，vに対応づけられている連句頂点ベクトルを$(v_1,\ldots,v_n)$とする．$v_i(1\leqq i\leqq n)$における適用規則をR_iとし，G'の規則の組$(R_1,\ldots,R_n)$を派生するようなGの規則$\rho:A\rightarrow f[A_1,\ldots,A_s]$が存在すると仮定する（条件[N1]より，このような規則ρは高々一つしか存在しない）．ρを非終端規則とする．各h$(1\leqq h\leqq s)$について，$v_1,\ldots,v_n$の子頂点で，$A_h^{[j]}(1\leqq j\leqq d(A_h))$をラベルにもつ頂点を$u_{hj}$とするとき（条件[f3]より，このような頂点は必ず存在し唯一つに定まる），$U_h=(u_{h1},\ldots,u_{hd(ah)})$は深さkの連句頂点ベクトルである．

また，上の各頂点vに対して，次の(a),(b)を行って得られる木は，tに対応する高さkの導出木プレフィクスである．

(a)ρが非終端規則のとき，各h$(1\leqq h\leqq s)$について，vの第h子として，新しい頂点u_hを追加する．但し，u_hには上の連句頂点ベクトルU_hを対応づける．

(b)ρが終端規則のとき，終端記号列の組f[]をラベルとしてもつ頂点を，vの唯一の子頂点として加える．

[C3]連句頂点ベクトルおよび導出木プレフィクスは上の[C1],[C2]で定義されるもの以外に存在しない． □

tをG'におけるαの導出木とし，k_tをtの高さとするとき，上の定義から，tに対応する高さk_tのGにおける導出木プレフィクスは，Gにおけるαの導出木である．

4．構文解析アルゴリズム

Gを任意のmcfgとし，Gから派生するcfgをG'とする．G'があいまいでない[1]と仮定する．以下で述べるアルゴリズムPARSE(α)では，与えられた系列αに対し，まず，G'に対して（あいまいでないcfgに対する構文解析法を用いて）αを構文解析し，その結果得られる導出木tに対して，tに対応するGにおける導出木プレフィクスを根頂点からトップダウンに求めていく．以下にアルゴリズムを示す．

［手続きPARSE(α)］

（入力）α:終端記号列．

（出力）$\alpha\in L(G)$のときGにおけるαの導出木，

$\alpha\notin L(G)$のとき"NO".

（手続き）次の(1)～(3)を順に行う．

(1)αをcfg G'に対して構文解析し，その結果が$\alpha\notin L(G')$であれば，"NO"を出力して停止．$\alpha\in L(G')$のとき，得られた導出木をtとし，tの高さをk_tとする．

(2)k=0からk_tまで順に，以下の(2a)を行う．

(2a)導出木プレフィクスの定義に従って，（k≧1のときは高さk-1の導出木プレフィクスを用いて，）tに対応するGにおける高さkの導出木プレフィクスを求める．もしそれが存在しなければ，"NO"を出力して停止する．

(3)Gにおけるαの導出木が得られたのでそれを出力する．

□

PARSE(α)の時間計算量を評価する．次の性質が成り立つ．

［性質2］G'をあいまいでないcfgとし，$\alpha\in L(G')$とする．G'におけるαの導出木の頂点の個数は$O(n)$ $(n=|\alpha|)$である．（証明は容易） □

G'があいまいでないという仮定から，PARSEのステップ(1)は$O(n^2)(n=|\alpha|)$で実行できる[1]．ステップ(2a)において，高さk-1の導出木プレフィクスから，深さkの連句頂点ベクトルおよび高さkの導出木プレフィクスを求める操作は，tにおける深さk-1とkの頂点の個数の和のオーダで求められる．従って，ステップ(2)全体は，tの頂点の個数のオーダで実行できる．これは性質2より$O(n)$である．以上より，上のPARSEの時間計算量は$O(n^2)$である．

文　献

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

(2)C.J. Pollard: "Generalized phrase structure grammars, head grammars, and natural language", Ph.D. dissertation, Stanford University (1984).

(3)嵩,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), **J71-D**, 5, pp.758-765 (昭63-05).

(4)嵩,関,藤井:"Head Languageおよび多重文脈自由言語の所属問題",信学論(D),**J71-D**,6, pp.935-941 (昭63-06).