Title	Context-Free GrammarsのAssociate Languagesの生成する Full Semi-AFLについて (オートマトン理論および言語理論の新展開)
Author(s)	田中, 秀尚
Citation	数理解析研究所講究録 (1976), 270: 14-30
Issue Date	1976-04
URL	http://hdl.handle.net/2433/105917
Right	
Type	Departmental Bulletin Paper
Textversion	publisher

数理解析研究所講究録
270巻 1976年 14-30

context-free grammarsのassociate languagesの生成するfull semi-AFLについて

東大 理 田中 秀尚

§1. 序

context-free grammarsの拡張として様々なregurated rewriting systems(例えば, matrix grammars, control grammars, programmed grammars など)が考案されており, 互いに密接な関係を持っている。本稿では matrix languages 全体 $\mathcal{M}^{\lambda}$ の研究手段として, context-free grammars の associate languages全体 $\mathcal{A}$ の生成する full semi-AFL $\hat{\mathcal{F}}(\mathcal{A})$ を考察することにしよう。

<u>定義 1.</u> $G=(V_N, \Sigma_T, X_0, P)$ を context-free grammar (以後 cfg と略記) とし, P (productionsの集合) を alphabetとみなすことにしよう。

$\alpha \in P^*$, $u, v \in (V_N \cup \Sigma_T)^*$ とするとき, $u \overset{\alpha}{\underset{G}{\Rightarrow}}{}^* v$ (αは, uから v への derivation の control word である) を次の

ように定義する。

(i)　任意の $(V_N \cup \Sigma_T)^*$ の元 u に対して，$u \xRightarrow[G]{\lambda}{}^* u$。

但し，λ は empty word を示す。

(ii)　$u, v \in (V_N \cup \Sigma_T)^*$, $\pi = X \rightarrow z \in P$ とすると，

$uXv \xRightarrow[G]{\pi}{}^* uzv$。

(iii)　$u \xRightarrow[G]{\alpha}{}^* v$ かつ $v \xRightarrow[G]{\beta}{}^* w$ ならば，$u \xRightarrow[G]{\alpha\beta}{}^* w$。

G の associate language $A(G)$ (Moriya, 1973) を次で定義する。

$$A(G) = \{ \alpha \in P^* \mid \exists w \in \Sigma_T^*,\ X_0 \xRightarrow[G]{\alpha}{}^* w \}.$$

各 $n \geqq 1$ に対して $\mathcal{A}(n)$ を

$$\mathcal{A}(n) = \{ A(G) \mid G = (V_N, \Sigma_T, X_0, P) \text{ は cfg で, } \# V_N \leqq n \}$$

と定義し，$\mathcal{A} = \bigcup_{n=1}^{\infty} \mathcal{A}(n)$ とおこう。

本稿では，control grammar とは，cfg $G = (V_N, \Sigma_T, X_0, P)$ と，P上の language C との対 (G, C) を意味することとする。(G, C) によって生成される language $L(G, C)$ は

$$L(G, C) = \{ w \in \Sigma_T^* \mid \exists \alpha \in C,\ X_0 \xRightarrow[G]{\alpha}{}^* w \}$$

で定義する。

一般に，$\mathcal{L}$ を languages の family とするとき，$\mathcal{G}^{\lambda}(\mathcal{L})$, $\mathcal{G}(\mathcal{L})$ をそれぞれ次のように定義する。

$$\mathcal{G}^{\lambda}(\mathcal{L}) = \{ L(G, C) \mid C \in \mathcal{L} \},$$

$$\mathcal{G}(\mathcal{L}) = \{ L(G,C) \mid C \in \mathcal{L},\ G \text{ は } \lambda\text{-free cfg} \}。$$

regular languages全体を$\mathcal{R}$，matrix languages全体を$\mathcal{M}^\lambda$と書くことにすれば，よく知られているように，$\mathcal{M}^\lambda = \mathcal{G}^\lambda(\mathcal{R})$となる（Salomaa [2], 1970）。

§2. Q-automataと$\mathcal{Q}^\lambda$

$\hat{\mathcal{G}}(\mathcal{A})$を受理するautomataとして，Q-automataを定義しよう。N，Zでそれぞれ自然数全体，整数全体を表わすこととする。又，一般に，二項関係ρに対して，ρ^*で，ρのreflexive transitive closureを表わすことにする。

定義2.　Q-automatonとは次のような六つ組$M = (n, K, \Sigma, q_0, F, P)$である。

(i)　nは正整数で，Mのdegreeと呼ばれる。

(ii)　KとΣとはalphabetsで，$K \cap \Sigma = \phi$。

(iii)　$q_0 \in K$，$F \subseteq K$。

(iv)　Pは$K \times (\Sigma \cup \{\lambda\}) \times K \times N^n \times N^n$の有限部分集合。Pが特に$K \times \Sigma \times K \times N^n \times N^n$の部分集合であるとき，Mを$\lambda$-freeと呼ぶ。

$K \times \Sigma^* \times N^n$上の二項関係$\underset{M}{\Rightarrow}$を次のように定義する。

$(p, a, q, x, y) \in P$，$w \in \Sigma^*$，$z \in N^n$のとき，$z \geqq x$ならば，

$(p, aw, z) \underset{M}{\Rightarrow} (q, w, z-x+y)$。

Q-automaton M の受理する language $L(M)$ は次で定義する。

$$L(M) = \{ w \in \Sigma^* \mid \exists p \in F, (q_0, w, 0_n) \underset{M}{\overset{*}{\Rightarrow}} (p, \lambda, 0_n) \}。$$

ここで $0_n = (0, 0, \cdots, 0) \in N^n$。

各 $n \geq 1$ に対して，degree n の（λ-free）Q-automata で受理される languages 全体を $\mathcal{Q}^\lambda(n)$（$\mathcal{Q}(n)$）で表わし，$\mathcal{Q}^\lambda = \bigcup_{n=1}^{\infty} \mathcal{Q}^\lambda(n)$，$\mathcal{Q} = \bigcup_{n=1}^{\infty} \mathcal{Q}(n)$ と書くことにしよう。

定義 3.　$\mathcal{L}, \mathcal{L}_1, \mathcal{L}_2$ を languages の families としよう。

$$\mathcal{L} + (\lambda) = \mathcal{L} \cup \{ L \cup \{\lambda\} \mid L \in \mathcal{L} \}。$$

$H_p(\mathcal{L}) = \{ h(L) \mid L \in \mathcal{L}$, h は length-preserving homo.$\}$。

$H_d(\mathcal{L}) = \{ h(L) \mid L \in \mathcal{L}$, h は decreasing homo.$\}$。

$$\mathcal{L}_1 \wedge \mathcal{L}_2 = \{ L_1 \cap L_2 \mid L_1 \in \mathcal{L}_1, L_2 \in \mathcal{L}_2 \}。$$

各 $n \geq 1$ に対して，$\wedge^n \mathcal{L}$ を次のように定義する。

$$\wedge^1 \mathcal{L} = \mathcal{L}, \quad \wedge^{m+1} \mathcal{L} = (\wedge^m \mathcal{L}) \wedge \mathcal{L}。$$

$\mathcal{L}$ を含む最小の（full）semi-AFL を $\mathcal{S}(\mathcal{L})$（$\hat{\mathcal{S}}(\mathcal{L})$）で表わす。特に，$\mathcal{L} = \{L\}$ のとき，$\mathcal{S}(L)$（$\hat{\mathcal{S}}(L)$）と略記し，L を $\mathcal{S}(L)$（$\hat{\mathcal{S}}(L)$）の（full）m-generator と呼ぶ。

$\mathcal{L}$ を含んで intersection で閉じている最小の（full）semi-AFL は $\bigcup_{n=1}^{\infty} H_p(\wedge^n \mathcal{S}(\mathcal{L}))$（$\bigcup_{n=1}^{\infty} H_d(\wedge^n \hat{\mathcal{S}}(\mathcal{L}))$）で表わされる。（[3], [4] 参照）

<u>定理 1.</u>　各 $n \geqq 1$ に対して，

$$\mathcal{L}(n) = H_p(\Lambda^n \mathcal{L}(1)), \quad \mathcal{L}^\lambda(n) = H_d(\Lambda^n \mathcal{L}(1))。$$

$\mathcal{L}(n)$は semi-AFL，$\mathcal{L}^\lambda(n)$は full semi-AFLである。

<u>定理 2.</u>　D_1を 1-Dyck language とすると，

$$\mathcal{L}(1) = \mathcal{F}(D_1)。$$

<u>系 1.</u>　各 $n \geqq 1$ に対して，

$$\mathcal{L}(n) = H_p(\Lambda^n \mathcal{F}(D_1)), \quad \mathcal{L}^\lambda(n) = H_d(\Lambda^n \mathcal{F}(D_1))。$$

<u>系 2.</u>　$\mathcal{L}$ ($\mathcal{L}^\lambda$) は D_1 を含んで intersectionで閉じている最小の (full) semi-AFLである。

<u>定義 4.</u>　Σ上の languages L_1, L_2 に対して，

$$\mathrm{Shuffle}(L_1, L_2) = \{x_1 y_1 \cdots x_n y_n \mid n \geqq 1,\ x_1 \cdots x_n \in L_1,\ y_1 \cdots y_n \in L_2,\ \forall x_i, y_i \in \Sigma^*\}$$

と定義する。このとき，$L_1 \subseteq \Sigma_1^*$, $L_2 \subseteq \Sigma_2^*$, $\Sigma_1 \cap \Sigma_2 = \phi$ とすると，

$$H_p(\mathcal{F}(L_1) \wedge \mathcal{F}(L_2)) = \mathcal{F}(\mathrm{Shuffle}(L_1, L_2)),$$

$$H_d(\mathcal{F}(L_1) \wedge \mathcal{F}(L_2)) = H_d(\hat{\mathcal{F}}(L_1) \wedge \hat{\mathcal{F}}(L_2)) = \hat{\mathcal{F}}(\mathrm{Shuffle}(L_1, L_2))$$

となる (Ginsburg, Greibach [4], 1970)。　この結果を利

用して，各 $\mathcal{L}(n)$ の m-generator を求めることができる。

各 $n \geq 1$ に対して，$D_1^{(n)}$ を cfg

$$G_n = (\{X\}, \{a_n, b_n\}, X, \{X \to XX,\ X \to a_n X b_n,\ X \to \lambda\})$$

で生成される 1-Dyck language とし，language $Q(n)$ を，

$$Q(1) = D_1^{(1)},$$

$$Q(m+1) = \mathrm{Shuffle}(Q(m), D_1^{(m+1)})$$

で定義する。

<u>系3.</u>　各 $n \geq 1$ に対して，

$$\mathcal{L}(n) = \mathcal{S}(Q(n)),\quad \mathcal{L}^{\lambda}(n) = \hat{\mathcal{S}}(Q(n))。$$

$\mathcal{L}(n) \supseteq \mathcal{A}(n)$ 及び $Q(n) \in \mathcal{S}(\mathcal{A}(n)) + (\lambda)$ を示すことによって次の定理を証明することができる。

<u>定理3.</u>　各 $n \geq 1$ に対して，

$$\mathcal{L}(n) = \mathcal{S}(\mathcal{A}(n)) + (\lambda),\quad \mathcal{L}^{\lambda}(n) = \hat{\mathcal{S}}(\mathcal{A}(n))。$$

<u>系4.</u>　$\mathcal{L} = \mathcal{S}(\mathcal{A}) + (\lambda),\quad \mathcal{L}^{\lambda} = \hat{\mathcal{S}}(\mathcal{A})$。

次に $\mathcal{L}^{\lambda}$ に含まれている languages の例をいくつかあげよう。

<u>定理4.</u>　各 $n \geqq 1$ に対して，

$$A(n) = \{ x c x^R \mid x \in (a^* b)^{n-1} a^* \},$$

$$A'(n) = \{ a^{k_1} b \cdots b a^{k_n} c a^{r_n} b \cdots b a^{r_1} \mid \forall i,\ k_i \geqq r_i \geqq 0 \},$$

$$B(n) = \{ x c x \mid x \in (a^* b)^{n-1} a^* \},$$

$$B'(n) = \{ a^{k_1} b \cdots b a^{k_n} c a^{r_1} b \cdots b a^{r_n} \mid \forall i,\ k_i \geqq r_i \geqq 0 \}$$

とおく。各 $n \geqq 1$ に対して，

$$A(n+1),\ A'(n+1),\ B(n+1),\ B'(n+1) \in \mathcal{L}(n+1) - \mathcal{L}(n)。$$

<u>系5.</u>　$\mathcal{L}(1) \subset \mathcal{L}(2) \subset \cdots \subset \mathcal{L}(n) \subset \cdots \subset \mathcal{L}$ は，真の hierarchy である。

<u>系6.</u>　$\{ x c x^R \mid x \in \{a, b\}^* \}$ や $\{ x c x \mid x \in \{a, b\}^* \}$ は $\mathcal{L}$ に含まれない。

私達は定理4が $\mathcal{L}^\lambda(n)$ に対しても成立すると予想しているが，証明はできていない。

<u>例1.</u>　各 $i = 1, 2$ に対して，degree 2 の Q-automaton $M_i = (2, \{q_0, q_1, q_2, q_3\}, \{a, b, c, d, e\}, q_0, \{q_3\}, P_i)$ を次のように定義しよう。

$$P_1 = \{ (q_0, a, q_1, (0,0), (1,0)), (q_1, b, q_1, (1,0), (0,1)),$$

$(q_1, c, q_2, (0,0), (1,0))$, $(q_2, d, q_2, (0,1), (1,0))$,

$(q_2, a, q_1, (0,0), (1,0))$, $(q_2, e, q_3, (2,0), (0,0))$,

$(q_3, e, q_3, (2,0), (0,0))\}$,

$P_2 = \{(q_0, a, q_1, (0,0), (1,0)), (q_1, b, q_1, (1,0), (0,1))$,

$(q_1, c, q_2, (0,0), (0,0)), (q_2, d, q_2, (0,1), (2,0))$,

$(q_2, a, q_1, (0,0), (0,0)), (q_2, e, q_3, (1,0), (0,0))$,

$(q_3, e, q_3, (1,0), (0,0))\}$。

例えば，$L(M_1)$ の元 $abcdab^3cd^2ab^3cd^4e^3$ の計算過程，及び $L(M_2)$ の元 $abcdab^2cd^2ab^4cd^2abcdabcd^3e^{10}$ の計算過程はそれぞれ図1，2に示されている。

図 1.

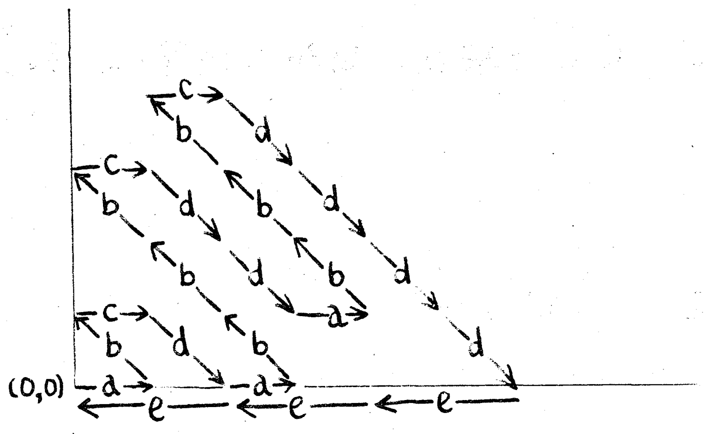
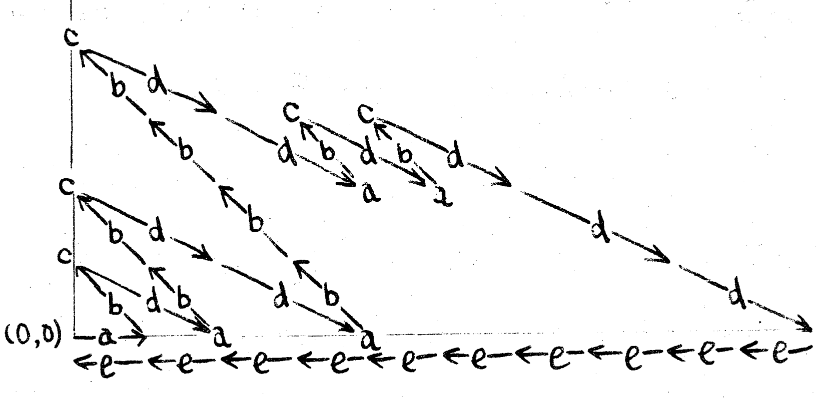


図 2.



$\{a,b,c,d,e\}$ 上の Parikh mapping Φ を $\Phi(a)=(1,0,0,0,0)$, $\Phi(b)=(0,1,0,0,0)$, $\Phi(c)=(0,0,1,0,0)$, $\Phi(d)=(0,0,0,1,0)$, $\Phi(e)=(0,0,0,0,1)$ と定義すると,

$$\Phi(L(M_1)) = \{(n,m,n,m,n) \mid n \geqq 1,\ n^2 \geqq m \geqq 1\},$$

$$\Phi(L(M_2)) = \{(n,m-1,n,m-1,m) \mid n \geqq 1,\ 2^n \geqq m \geqq 1\}。$$

従って, $\{a^n b^m \mid n \geqq 1,\ n^2 \geqq m \geqq 1\}$ や $\{a^n b^m \mid n \geqq 1,\ 2^n \geqq m \geqq 1\}$ は $\mathcal{L}^\lambda(2)$ に含まれる。

§3. $\mathcal{L}^\lambda$ と $\mathcal{M}^\lambda$

$\mathcal{L}^\lambda$ と $\mathcal{M}^\lambda$ が密接な関係を持っていることは, 次のような事実(定理)及び未解決の問題の同等性(系)から知ることができる。

<u>定理 5.</u> $\mathcal{M}^\lambda \supset \mathcal{L}^\lambda$。

<u>系 7.</u> $\mathcal{M}^\lambda$ に対する emptiness problem と, $\mathcal{L}^\lambda$ に対する emptiness problem とは同等である。

<u>定理 6.</u> $\mathcal{M}^\lambda = \xi^\lambda(\mathcal{L}^\lambda) = \xi(\mathcal{L}^\lambda)+(\lambda)$。更に, 各 matrix language L に対して, 適当な λ-free cfg G と $\mathcal{L}^\lambda$ の元 C をとることができて, $L-\{\lambda\} = L(G,C)$ かつ, G は $X \to Y$ (X,

Y は nonterminals）のような production を持たない。

<u>系 8.</u>　context-sensitive languages 全体を $\mathcal{CS}$ としよう。問題 "$\mathcal{M}^\lambda \subseteq \mathcal{CS}$？" と問題 "$\mathcal{Q}^\lambda \subseteq \mathcal{CS}$？" とは同等である。

Salomaa は [5]（1970）で，$\mathcal{M}^\lambda$ は nonregular one-letter language を含まないという予想を述べている。$\mathcal{C}$，$\mathcal{B}$ をそれぞれ commutative languages 全体，bounded languages 全体としよう。

<u>定理 7.</u>　$\mathcal{M}^\lambda \cap \mathcal{C} = \mathcal{Q}^\lambda \cap \mathcal{C}$，　$\mathcal{M}^\lambda \cap \mathcal{B} = \mathcal{Q}^\lambda \cap \mathcal{B}$。

<u>系 9.</u>　$\mathcal{M}^\lambda$ が nonregular one-letter language を含まないということと，$\mathcal{Q}^\lambda$ が nonregular one-letter language を含まないということとは同値である。

§4.　commutative automata と $\mathcal{L}_c^\lambda$

系 7〜9 に述べられている問題は残念ながら未解決である。Q-automata を少し変形して得られる $\mathcal{Q}^\lambda$ の subfamily $\mathcal{L}_c^\lambda$ について，これらの問題を考えてみよう。Q-automata の記憶部は，非負整数の vector であったが，この非負条件を取

り去った automata を次のように定義しよう。

定義 5.　commutative automaton とは次のような六つ組 $M=(n, K, \Sigma, q_0, F, P)$ である。

(i)　n は正整数で，M の degree と呼ばれる。

(ii)　K と Σ とは alphabets で，$K \cap \Sigma = \phi$。

(iii)　$q_0 \in K$，$F \subseteq K$。

(iv)　P は $K \times (\Sigma \cup \{\lambda\}) \times K \times Z^n$ の有限部分集合。

特に $P \subseteq K \times \Sigma \times K \times Z^n$ のとき，M を λ-free と呼ぶ。

$K \times \Sigma^* \times Z^n$ 上の二項関係 $\underset{M}{\Rightarrow}$ を次のように定義する。

$(p, a, q, x) \in P$，$w \in \Sigma^*$，$y \in Z^n$ のとき，

$$(p, aw, y) \underset{M}{\Rightarrow} (q, w, x+y)。$$

M の受理する language $L(M)$ は次で定義される。

$$L(M) = \{ w \in \Sigma^* \mid \exists p \in F,\ (q_0, w, 0_n) \underset{M}{\overset{*}{\Rightarrow}} (p, \lambda, 0_n) \}。$$

各 $n \geq 1$ に対して，degree n の (λ-free) commutative automata で受理される languages 全体を $\mathcal{L}_c^\lambda(n)$ ($\mathcal{L}_c(n)$) で表わし，$\mathcal{L}_c^\lambda = \bigcup_{n=1}^\infty \mathcal{L}_c^\lambda(n)$，$\mathcal{L}_c = \bigcup_{n=1}^\infty \mathcal{L}_c(n)$ とおく。

定理 8.　各 $n \geq 1$ に対して，

$$\mathcal{L}_c(n) = H_p(\wedge^n \mathcal{L}_c(1)),\quad \mathcal{L}_c^\lambda(n) = H_d(\wedge^n \mathcal{L}_c(1))。$$

定理9. $\mathcal{L}_c(1) = \mathcal{S}(O)$。ここで，

$$O = \{ w \in \{a, b\}^* \mid N_a(w) = N_b(w) \},$$

$N_e(w)$ は w に現われる symbol e の個数を示す。

系10. 各 $n \geq 1$ に対して，

$$\mathcal{L}_c(n) = H_p(\wedge^n \mathcal{S}(O)), \quad \mathcal{L}_c^\lambda(n) = H_d(\wedge^n \mathcal{S}(O))。$$

系11. $\mathcal{L}_c$ $(\mathcal{L}_c^\lambda)$ は O を含んで intersection で閉じている最小の (full) semi-AFL である。

定義6. 各 $n \geq 1$ に対して，dimension n の origin-crossing language $O(n)$ を

$$O(n) = \{ w \in \{a_1, b_1, \cdots, a_n, b_n\}^* \mid \forall i, N_{a_i}(w) = N_{b_i}(w) \}$$

で定義する (Fischer et al., 1968)。

系12. 各 $n \geq 1$ に対して，

$$\mathcal{L}_c(n) = \mathcal{S}(O(n)), \quad \mathcal{L}_c^\lambda(n) = \hat{\mathcal{S}}(O(n))。$$

定理10. semilinear property を持つ commutative languages 全体を $\mathcal{C}_{SL}$ と表わすことにすれば，

$$\mathcal{L}_c = \mathcal{S}(\mathcal{C}_{SL}), \quad \mathcal{L}_c^\lambda = \hat{\mathcal{S}}(\mathcal{C}_{SL})。$$

従って，$\mathcal{L}_C^{\lambda}$ に対する emptiness, finiteness problems は帰納的に可解となり，かつ又，$\mathcal{L}_C^{\lambda}$ は nonregular one-letter language を含まない（Ginsburg, Spanier [7], 1971）。

<u>定理11.</u>　各 $n \geqq 1$ に対して，$\mathcal{L}_C^{\lambda}(n) = \mathcal{L}_C(n)$。

<u>系13.</u>　$\mathcal{L}_C^{\lambda} = \mathcal{L}_C$，即ち，$\hat{\mathscr{F}}(\mathcal{E}_{SL}) = \mathscr{F}(\mathcal{E}_{SL})$。

deterministic context sensitive languages 全体を $\mathcal{CS}^d$ と表わすことにしよう。$\mathcal{CS}^d$ は intersection で閉じている AFL で，context-free languages（cfl's と略記）全体 $\mathcal{F}$ を含んでいる。O は cfl であるから次の定理が成立する。

<u>定理12.</u>　$\mathcal{L}_C^{\lambda} = \hat{\mathscr{F}}(\mathcal{E}_{SL}) \subseteq \mathcal{CS}^d$。

$\mathcal{L}_C^{\lambda}$を利用すると，[7] で残された「$\mathcal{E}_{SL}$ によって生成される full AFL $\hat{\mathscr{F}}(\mathcal{E}_{SL})$ は full principal AFL であるか。」という問題を解くことができる。$A(n), A'(n), B(n), B'(n)$ は定理4で定義された languages とする。

<u>定理13.</u>　各 $n \geqq 1$ に対して，

$$C(n) = \{(a^k c)^n a^k \mid k \geq 0\}$$

とおく。各 $n \geq 1$ に対して，

$$A(n+1),\ A'(n+1),\ B(n+1),\ B'(n+1),\ C(n+1) \in \hat{\mathcal{L}}^{\lambda}_{C}(n+1) - \hat{\mathcal{L}}^{\lambda}_{C}(n)。$$

languages の family $\mathcal{L}$ に対して，$\mathcal{L}$を含んでいる（$\mathcal{L}$を含んで λ-free substitution で閉じている）最小の AFL を $\mathcal{F}(\mathcal{L})$（$\mathcal{F}_{\sigma}(\mathcal{L})$）で表わすことにしよう。B(n), C(n) は次のような強い性質を持っている。language L が 性質 (I3)（(I2)）を持っているとき，L を type 3（type 2）の invariant language と呼ぶ（[8] の Part 3 参照）。

(I3) 任意の semi-AFL $\mathcal{L}$ に対して，L が $\mathcal{F}(\mathcal{L})$ に含まれれば，L は $\mathcal{L}$ に含まれる。

(I2) 任意の semi-AFL $\mathcal{L}$ に対して，L が $\mathcal{F}_{\sigma}(\mathcal{L})$ に含まれれば，L は $\mathcal{L}$ に含まれる。

A(n), B(n), C(n) は type 3 の invariant language であり，更に，$n \geq 2$ であれば，B(n), C(n) は type 2 の invariant language である。従って次の系が得られる。

<u>系 14.</u> $\hat{\mathcal{F}}(\mathcal{C}_{SL})$ は full principal AFL ではない。

<u>系 15.</u> 次の包含関係は真の包含関係である。

$$
\begin{array}{ccccccccc}
\mathcal{L}_C^\lambda(1) & \subset & \mathcal{L}_C^\lambda(2) & \subset \cdots \subset & \mathcal{L}_C^\lambda(n) & \subset \cdots \subset & \mathcal{L}_C^\lambda & (= \hat{\mathcal{F}}(\mathcal{C}_{SL})) \\
\cap & & \cap & & \cap & & \cap & \\
\hat{\mathcal{F}}(\mathcal{L}_C^\lambda(1)) & \subset & \hat{\mathcal{F}}(\mathcal{L}_C^\lambda(2)) & \subset \cdots \subset & \hat{\mathcal{F}}(\mathcal{L}_C^\lambda(n)) & \subset \cdots \subset & \hat{\mathcal{F}}(\mathcal{L}_C^\lambda) & (= \hat{\mathcal{F}}(\mathcal{C}_{SL})) \\
\cap & & \cap & & \cap & & \cap & \\
\hat{\mathcal{F}}_\cap(\mathcal{L}_C^\lambda(1)) & \subset & \hat{\mathcal{F}}_\cap(\mathcal{L}_C^\lambda(2)) & \subset \cdots \subset & \hat{\mathcal{F}}_\cap(\mathcal{L}_C^\lambda(n)) & \subset \cdots \subset & \hat{\mathcal{F}}_\cap(\mathcal{L}_C^\lambda) & (= \hat{\mathcal{F}}_\cap(\mathcal{C}_{SL}))。
\end{array}
$$

但し，$\mathcal{L}_C^\lambda$がAFLでないことは，$(c(1)b)^* \notin \mathcal{L}_C^\lambda$からわかる。

最後に，$\mathcal{L}_C^\lambda$は，$\mathcal{Q}$のsubfamilyであることに注意しよう。

定理14. 各$n \geq 1$に対して，$\mathcal{Q}(2n) \supset \mathcal{L}_C^\lambda(n)$。

系16. $\mathcal{Q} \supset \mathcal{L}_C^\lambda$。この包含関係は，$\mathcal{Q}$がsemilinear propertyを持たない(例1)から，真の包含関係である。

§5. まとめ

系7〜9で述べられている$\mathcal{M}^\lambda$に対する問題は，$\mathcal{Q}^\lambda$に対する問題と制限しても同等であること(§3)，及び，$\mathcal{Q}^\lambda$のsubfamily $\mathcal{L}_C^\lambda$に対する問題と制限すれば解決されること(§4)をみてきた。$\mathcal{L}_C^\lambda$に対する結果は，Cremers と Mayer ([9], [10])によって定義されたgeneralized unordered vector languages全体$\mathcal{UV}_\infty^\lambda$にまで拡張できることを注意しておこう。

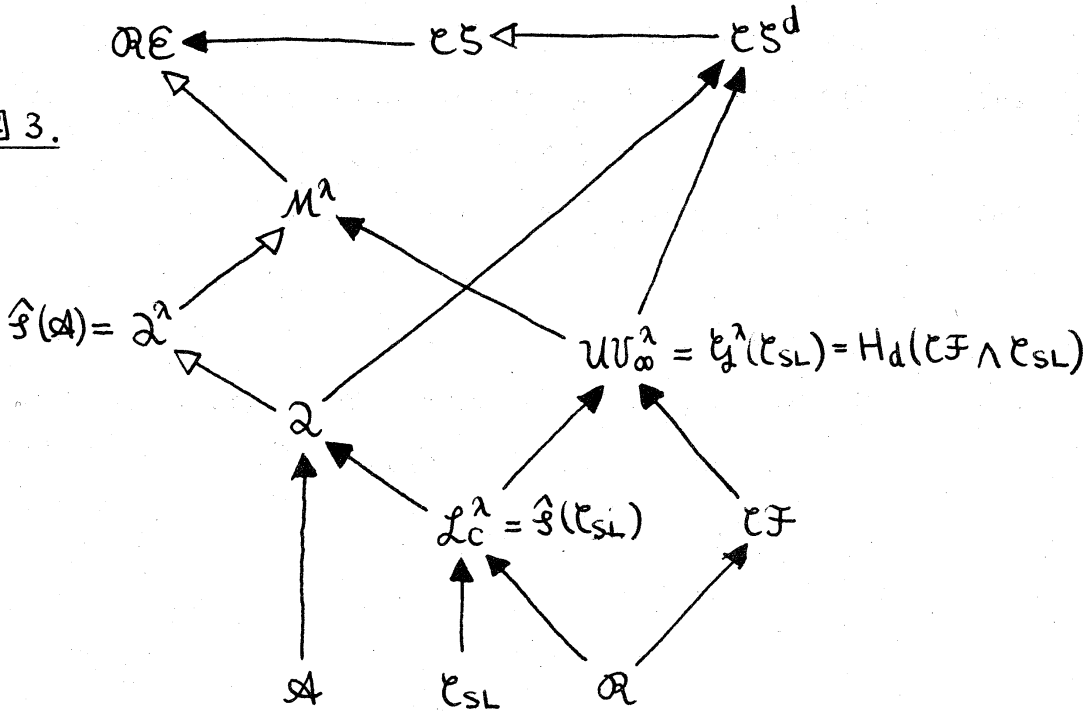
定理15. $\mathcal{UV}_\infty^\lambda = \mathcal{G}^\lambda(\mathcal{C}_{SL}) = \mathcal{G}(\mathcal{C}_{SL}) + (\lambda) = \mathcal{UV}_\infty + (\lambda)$。更

に，$\mathcal{UV}_\infty^\lambda$ の元 L に対して，適当な λ-free cfg G と $\mathcal{C}_{SL}$ の元 C をとることができて，$L-\{\lambda\}=L(G,C)$ かつ G は $X\to Y$（X, Y は G の nonterminals）のような production を持たない。

<u>系 17.</u>　$\mathcal{UV}_\infty^\lambda \subseteq \mathcal{CS}^d$。

本稿でふれた families の包含関係は，図 3 のように表わすことができる。

<u>図 3.</u>



$\mathcal{RE}$ は recursively enumerable languages 全体をさしている。←— は，包含関係が真であることを示し，◁— は，包含関係が真であるかどうか未解決であることを示している。

本稿では省略した定理の証明等の詳細は筆者の論文[8]を参照されたい。

参考文献

[1] E. Moriya, Associate languages and derivational complexity of formal grammars and languages, Information and Control 22 (1973), 139-162.

[2] A. Salomaa, Periodically time-variant context-free grammars, Information and Control 17 (1970), 294-311.

[3] S. Greibach and S. Ginsburg, Multitape AFA, J. Assoc. Comput. Mach. 19 (1972), 193-221.

[4] S. Ginsburg and S. Greibach, Principal AFL, J. Comput. System Sci. 4 (1970), 308-338.

[5] A. Salomaa, On some families of formal languages obtained by regulated derivations, Ann. Acad. Sci. Fenn., Ser. AI. 479 (1970).

[6] P. C. Fischer, A. R. Meyer, and A. L. Rosenberg, Counter machines and counter languages, Math. Systems Theory 2 (1968), 265-283.

[7] S. Ginsburg and E. H. Spanier, AFL with the semilinear property, J. Comput. System Sci. 5 (1971), 365-396.

[8] 田中, Elementary methods for formal language theory and matrix languages, 東大修士論文.

[9] A. B. Cremers and O. Mayer, On matrix languages, Information and Control 23 (1973), 86-96.

[10] A. B. Cremers and O. Mayer, On vector languages, J. Comput. System Sci. 8 (1974), 158-166.