解　説

属性文法とその応用—IV

属性文法の複雑さ†

西野哲朗†† 中田育男†††

1. はじめに

本連載の第1回「属性文法の理論入門」では，属性文法の定義から始めて，基本的な属性評価アルゴリズムについて解説した．第1回を読まれた読者のなかには，そこで定義された種々の属性文法が，表現能力の点でどのように異なるのかということを，疑問に思われた方も多いと思う．そこで今回は，種々の属性評価アルゴリズムの計算能力や，属性文法の変換能力を計算量の観点から考察する．

第1回の冒頭でも述べたが，第1回は属性文法の基本公式集であり，今回は細則集（あるいは中級コース）である．今回の内容を理解していただければ，(1)属性文法理論の最も重要な事項をほぼ修得したことになり，(2)属性文法の応用を試みる際の基礎もでき，さらに，(3)参考文献などを読破することも比較的容易になるものと思われる．

ご存知のように，理論では破綻のない数学的形式化が大切なので，当該理論に馴染みのない方には，記述が抽象的で分かりにくいことが多い．しかし，実際に理論を整備している理論屋のほうも，数学的な言葉を使って表現したい具体的な内容がまず先にあるのが普通である．その意味で，今回の解説が抽象的で分かりにくいと感じられたら，具体例を用いてその定義の意味を確認していただきたい．そうすることにより，一つの定義が実質的に何を述べているかが分かるはずである．

そこで本稿では，具体例によって定義の意味が確認できるように配慮したつもりである．一部の読者にとっては，本稿を読むことは，推理小説を読む（あるいは暗号を解読する）に等しい知的興奮（あるいは苦痛？）をともなうことかもしれないが，是非謎解きを楽しんでいただきたいと思う．

2. 属性文法の静的複雑さ

頁数の都合上，本連載第1回の続きから話を始めるので，基本的な定義や記法については第1回を参照されたい．なお，本稿は属性文法の概略をできるだけコンパクトに述べることを目標としているので，本稿の内容の詳細については，たとえば文献 2)，18) などを参照されたい．

X, Y を属性文法のクラスとする．X 属性文法に対する属性評価アルゴリズムが，Y 属性文法に対する属性評価アルゴリズムの特殊な場合にあたるとき，X 属性文法を Y 属性文法の部分クラスと考える（たとえば，X=単純多重訪問，Y=絶対非循環など）．このようにして定義された属性文法のクラス階層を**図-1** に示す．図-1 では，以下

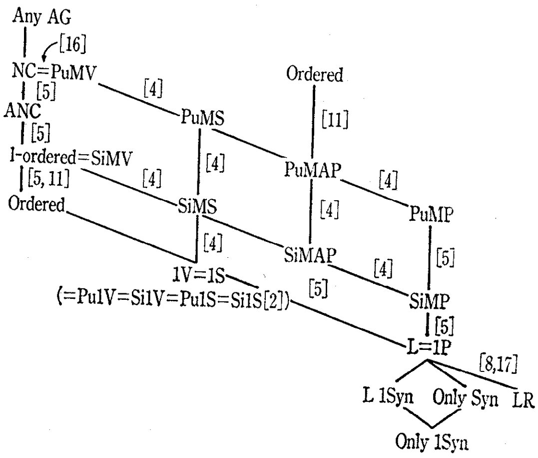


図-1　属性文法のクラス階層2),18)

† Attribute Grammars and Computational Complexity by Tetsuro NISHINO (School of Information Science, Japan Advanced Institute of Science and Technology, Hokuriku) and Ikuo NAKATA (Institute of Information Sciences and Electronics, University of Tsukuba).

†† 北陸先端科学技術大学院大学情報科学研究科
††† 筑波大学電子情報工学系

の略記法を用いている．NC：非循環，ANC：絶対非循環，Pu：純粋，Si：単純，M：多重，V：訪問，S：スイープ，A：交互，P：パス．

以下では，図-1中に現れる属性文法のクラスのうち，まだ定義を述べていないものについて説明する．まず最初に，応用上特に重要な 1-ordered および Ordered 属性文法について述べる．その前に，属性文法の静的性質を解析するときに便利な，グラフを用いた概念を定義する．

定義 2.1 $p: X_0 \rightarrow w_0X_1w_1 \cdots X_nw_n$ を，属性文法 G のプロダクションとする．

1. $O_p=\{a(X_j) \mid a \in A(X_j), 0 \leq j \leq n\}$ を p の属性生起全体の集合とする．p のプロダクショングラフ $PG(p)=(O_p, D_p)$ とは，O_p の各要素に対応したノードをもち，p において属性 b が属性 a に依存するとき，かつそのときに限り，a から b に向かう辺を（D_p の要素として）もつ有効グラフである．

2. R_j, $0 \leq j \leq n$ を $A(X_j)$ 上の任意の関係とする．$PG(p)[R_0, R_1, \cdots, R_n]$ で，有向グラフ $(O_p, D_p \cup D_0 \cup D_1 \cup \cdots \cup D_n)$ を表すものとする．ここで，a と b を結ぶ辺が D_j 内に存在するのは，$a=c(X_j)$, $b=d(X_j)$ かつ $(c, d) \in R_j$ のとき，かつそのときに限るものとする．

第1回，**2.**の属性文法 G_2 のプロダクショングラフを**図-2** に示す．

練習問題 1 第1回，**2.**の属性文法 G_2 について考える．$A(B)$ 上の関係として $R_1=\{(p, v)\}$ を考えるとき，有向グラフ $PG(p_3)[R_0, R_1]$ を書け．ただし，ここでは $R_0=\phi$ とする．

属性文法 G は，各非終端記号 X に対して $A(X)$ 上の線形順序 L_X が存在し，G の各プロダクション

$$p: X_0 \rightarrow w_0X_1w_1 \cdots X_nw_n$$

に対し，$PG(p)[L_{X_0}, L_{X_1}, \cdots, L_{X_n}]$ が非循環的なときに，**1-ordered** であるという．1-ordered 属性文法のクラスは，単純多重属性文法のクラスと一致することが知られている（図-1 参照）．

練習問題 2 第1回，**2.** の属性文法 G_2 は，1-ordered 属性文法であることを示せ．

Ordered 属性文法は 1-ordered 属性文法の部分クラスである．その定義は若干複雑なのでここでは省略する．詳細は文献 11) を参照されたい．与えられた属性文法が Ordered であるか否かの判定は，多項式時間で行えることが知られている[11]．Ordered 属性文法は，いくつかのコンパイラ生成系で実際に用いられている[2]．

最後に，まだ述べていない属性文法のクラスを箇条書きにする．

L 属性文法：任意の導出木上ですべての属性を，深さ優先左側順の1パスで評価できる属性文法のクラス．定義より明らかに，L属性文法のクラスは，単純1パス属性文法のクラスと一致する[13]．L属性文法においては，LL 構文解析と同時に属性評価が行えるので，導出木を必ずしも作る必要がない．

LR 属性文法：LR(k) 構文解析と同時に属性評価が行える属性文法のクラス[8),17]．

Only Syn 属性文法：各非終端記号に合成属性のみが付随している属性文法のクラス[3]．

L-1 Syn 属性文法：L かつ 1 Syn である属性文法のクラス[3]．ここで，**1 Syn 属性文法**とは，各非終端記号に合成属性が高々1個だけ付随している属性文法のことをいう（ただし，継承属性は何個付随していてもかまわない）[3]．

Only 1 Syn 属性文法：Only Syn かつ 1 Syn である属性文法のクラス[3]．

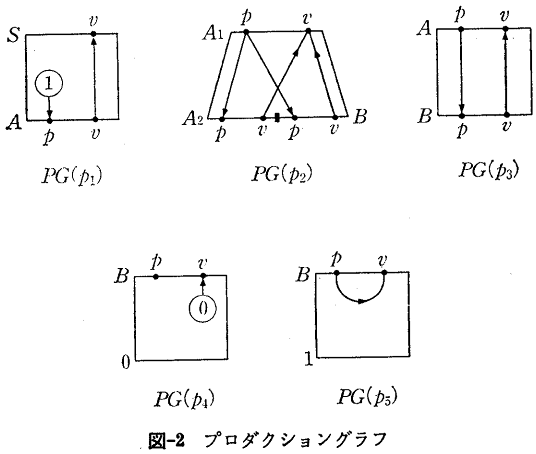


図-2 プロダクショングラフ

3. 属性文法に関する決定問題の複雑さ

本章では，与えられた属性文法が，ある性質をもつか否かを決定する問題の時間計算量について述べる．以下では，純粋多重戦略および，単純多重戦略属性文法の8つのクラスに関する，3種類の決定問題の時間計算量について述べる．X∈{訪問，スイープ，交互パス，パス} とする．

表-1　決定問題の複雑さ

<table>
<tr><th></th><th>問題A</th><th>問題B</th><th>問題C</th></tr>
<tr><td>SiMP　[1]</td><td rowspan="3">多項式時間</td><td rowspan="2">多項式時間</td><td>多項式時間</td></tr>
<tr><td>SiMAP　[9]</td><td rowspan="3">NP 完全</td></tr>
<tr><td>SiMS</td><td rowspan="2">NP 完全　(k>2)
多項式時間 (k=1)</td></tr>
<tr><td>SiMV</td><td>NP 完全</td></tr>
<tr><td>PuMP</td><td rowspan="2">EXP 完全</td><td rowspan="2">多項式時間</td><td rowspan="3">決定可能</td></tr>
<tr><td>PuMAP</td></tr>
<tr><td>PuMS</td><td>指数的下界</td><td rowspan="2">決定可能</td></tr>
<tr><td>PuMAV [16]</td><td>非循環性より自明</td><td>EXP 困難　[14]</td></tr>
</table>

注．[ ] 内に参考文献番号の指定なきものは，[4] 参照．

問題 A：与えられた属性文法が，純粋／単純多重X属性文法か否かを決定せよ．

問題 B：定められた整数 $k>0$ に対し，与えられた属性文法が純粋／単純 k-X 属性文法か否かを決定せよ．

問題 C：与えられた属性文法 G と整数 $k>0$ に対し，G が純粋／単純 k-X 属性文法か否かを決定せよ．

ここで，問題Cは以下の最適化問題Dに還元可能である．また，その逆も成り立つ．

問題 D：与えられた純粋／単純多重X属性文法 G に対し，G が純粋／単純 k-X 属性文法となるような最小の k を決定せよ．

これらの問題の難しさを**表-1** に示す．

練習問題 3　問題Cを解く多項式時間アルゴリズムが与えられれば，問題Dを多項式時間で解けることを示せ．また，その逆も成り立つことを示せ．

4. 属性文法の変換能力

本章では，いままでとは違った観点から得られる属性文法のクラス階層を紹介する．すなわち，属性文法の変換系としての能力と，それに対応して定義される属性文法のクラスについて述べる．そのためには，属性文法によって行われる変換を厳密に定義しなければならないので，以下ではまず意味領域という概念を導入し，第1回で述べた属性文法の定義を少し精密化して，ある意味領域上の属性文法という概念を定義する．

定義 4.1[3]　意味領域 Dom とは，2項組 (Val, Fun) のこととする．ここに，Val は属性値集合と呼ばれる集合族とし，Fun は，$f: V_1\times V_2\times\cdots\times V_m\to V_0$ なる形をした写像の集合とする．ただし，$0\leq i\leq m$ に対し $V_i\in\mathrm{Val}$ とする．Fun に属する写像を**意味関数**と呼ぶ．$m=0$ の場合は，f は V_0 の要素となる．

属性文法の理論でよく用いられる Dom の例としては，STRING（記号列全体の集合）や TREE（木全体の集合）などがある．また通常これらの意味領域は，意味関数として記号列の連接（または木の連接）などの関数を含んでいるものと仮定される．

次に，第1回で述べた属性文法の定義を精密化し，ある意味領域上の属性文法という概念を定義する．以下の定義においては，第1回の定義 3.2 と異なる部分を下線で示した．

定義 4.2　<u>意味領域 Dom=(Val, Fun) 上の</u>**属性文法** G とは，以下の1～3を満たす3項組 (G', A, F) のことをいう．

1. $G'=(N, T, P, Z)$ は既約な文脈自由文法であり，G の**基底文脈自由文法**と呼ばれる．

2. G' の各非終端記号 X に対し，交わりをもたない二つの有限集合 $I(X)$ と $S(X)$ が付随している．$I(X)$ の要素を**継承属性**，$S(X)$ の要素を**合成属性**という．X の属性全体の集合を $A(X)$ で表す．すなわち，$A(X)=I(X)\cup S(X)$ である．$A=\cup_{X\in N}A(X)$ と定義し，A を G の**属性集合**と呼ぶ．X の属性 a を $a(X)$ と表す．また，属性 a が取りうる値の集合を $V(a)$ と表す．

3. P の各プロダクション

$$p: X_0\to w_0X_1w_1\ \cdots\ X_nw_n\quad (n\geq 0)$$

に対し，

$$S(X_0)\cup I(X_1)\cup\ \cdots\ \cup I(X_n)$$

に含まれるすべての属性を定義するような，**意味規則**の集合 F_p が付随している．属性 $a(X_k)$ $(0\leq k\leq n)$ を定義する意味規則は，次のような形をしている．

$$a(X_k):=f(a_1(X_{i_1}),\cdots,a_m(X_{i_m}))\quad (*)$$

ただし，f は以下の型の <u>Fun に属する関数</u>とする．

$$V(a_1(X_{i_1}))\times\ \cdots\ \times V(a_m(X_{i_m}))\to V(a(X_k))$$

上の式(*)が成り立っているとき，$a(X_k)$ は p において $a_1(X_{i_1}),\cdots,a_m(X_{i_m})$ に**依存**しているという．$F=\cup_{p\in P}F_p$ と定義し，F を G の**意味規則集合**と呼ぶ．□

定義 4.3[4]　G を非循環属性文法とし，t を G の任意の導出木とする．$m_G(t)$ で G の導出木 t の根に付随した（唯一の）合成属性の値を表すことにする．G によって定められる**変換** T_G を以下のように定義する．

$T_G=\{(w,v)|w$ は t によって導出された終端記号列であり，かつ，$v=m_G(t)\}$

練習問題 4　第1回，**2.** の属性文法 G_2 においては，意味領域はどのように取られているか？　また T_{G2} はどのような集合となるか？

属性文法のクラス X と意味領域 D に対し，

$$T(X,D) = \{T_G|G\in X, G \text{ は } D \text{ 上の属性文法}\}$$

と定義する．X,Y を属性文法のクラスとするとき，X が Y よりも**強い変換能力をもたない**（$X\leq Y$ と表す）とは，すべての意味領域 D について，$T(X,D)\subseteq T(Y,D)$ となることと定義する．X と Y は，各意味領域 D 上で $T(X,D)=T(Y,D)$ となるとき，**同じ変換能力をもつ**という．次の定理が知られている．

定理 4.1[4]　$k>0$，$X\in$ {Visit, Sweep, AltPass, Pass} とする．このとき，純粋 k-X 属性文法のクラスと単純 k-X 属性文法のクラスは同じ変換能力をもつ．□

この定理により，たとえば，すべての意味領域 D に対し，T(k-PuV, D)＝T(k-SiV,D) が成り立つ☆．したがって，変換能力に対応した属性文法のクラス階層を考えるときには，純粋と単純を区別する必要がない．よって以下では，PuMV と SiMV をまとめて MV などと表すことにする．興味深いことに，変換能力に対応した属性文法のクラス階層は，**図-3** に示すように，k の大小に依存する．図-3 の階層図の証明は文献 4) を参照されたい．また，MV＝ANC の証明は文献 12) を参照されたい．

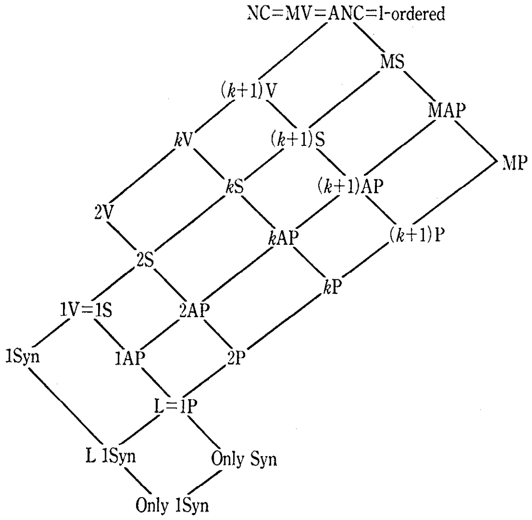


図-3　変換能力による階層[2),4)]

5. おわりに

本稿では頁数の関係で省略したが，ほかにもいろいろな属性文法理論の研究が行われている．たとえば，属性文法が生成する言語の複雑さに関する研究が行われており[6),7)]，それらの結果を応用して，自然言語に対する文法の生成能力なども研究されている[15)]．また，属性文法の動的複雑さに関しては，属性評価における大域的領域割当ての発見法に関する研究がある[10)]．これらの研究や他の研究の詳細については，文献 2) などを参照されたい．

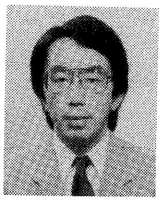
参考文献

1) Bochmann, G. V.: Semantic Evaluation from Left to Right, *CACM*, **19**, pp. 56-62 (1976).
2) Deransart, P., Jourdan, M. and Lorho, B.: Attribute Grammars—Definitions, Systems and Bibliography, *Lecture Notes in Computer Science* **323**, Springer (1988).
3) Engelfriet, J. and Filé, G.: The Formal Power of One-Visit Attribute Grammars, *Acta Informatica* **16**, pp. 275-302 (1981).
4) Engelfriet, J. and Filé, G.: Passes, Sweeps, and Visits, *Lecture Notes in Computer Science* **115**, pp. 193-207, Springer (1981).
5) Engelfriet, J. and Filé, G.: Simple Multi-Visit Attribute Grammars, *JCSS* **24**, pp. 283-314 (1982).
6) Engelfriet, J.: The Complexity of Languages Generated by Attribute Grammars, *SIAM J. Comput.* **15** (1986).
7) Ephremidis, S., Papadimitriou, C. H. and Sideris, M.: Complexity Characterizations of Attribute Grammar Languages, In: *Proceedings of Structure in Complexity Theory, Second Annual Conference, June 16-19, Cornel University, Ithaca, NY*, IEEE Computer Society Press, pp. 10-13 (1987).

☆ 2. で定義した略記法により，k-PuV は純粋 k 訪問 (pure k-visit) 属性文法のクラスを，k-SiV は単純 k 訪問 (simple k-visit) 属性文法のクラスを表していることに注意されたい．

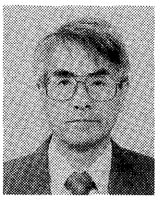
8) Jones, N.D. and Madsen, M.: Attribute-Influenced LR Parsing, *Lecture Notes in Computer Science*, Vol. 94, pp. 393-407, Springer (1980).
9) Jazayeri, M. and Walter, K.G.: Alternating Semantic Evaluator, *Proc. ACM 1975 Ann. Conf.*, pp. 230-234 (1975).
10) Katayama, T. and Sasaki, H.: Global Storage Allocation in Attribute Evaluation, *Proc. 13rd ACM Symp. on POPL* (1986).
11) Kastens, U.: Ordered Attribute Grammars, *Acta Informatica* **13**, pp. 229-256 (1980).
12) Kennedy, K. and Warren, S.K.: Automatic Generation of Efficient Evaluators for Attribute Grammars, *Proc. 3rd ACM Symp. on POPL*, pp. 32-49 (1976).
13) Lewis, P.M., Rosenkrantz, D.J. and Stearns, R.E.: Attributed Translations, *Journal of Computer and System Sciences*, Vol. 9, pp. 279-307 (1974).
14) Nishino, T.: The Intrinsically Exponential Complexity of the k-visit Property Problem for Attribute Grammars, In: S. Suzuki (Ed.) *Topology and Computer Science*, Kinokuniya pp. 473-486 (1987).
15) Nishino, T.: Relating Attribute Grammars and Lexical-Functional Grammars, *Information Sciences: An International Journal*, **66**, pp. 1-22 (1992).
16) Riis, H. and Skyum, S.: k-visit Attribute Grammars, *Math. Syst. Theory*, Vol. 15, pp. 17-28 (1981).
17) Sassa, M., Ishizuka, H. and Nakata, I.: A Contribution to LR Attributed Grammars, *Journal of Information Processing*, Vol. 8, pp. 196-206 (1985).
18) 佐々政孝：チュートリアル―属性文法，コンピュータソフトウェア，Vol. 3, pp. 73-91 (1986).

(平成6年1月26日受付)



西野　哲朗（正会員）

昭和34年生．昭和57年早稲田大学理工学部数学科卒業．昭和59年同大学院理工学研究科博士前期課程修了．同年日本アイ・ビー・エム(株)入社．昭和62年東京電機大学理工学部情報科学科助手．平成4年北陸先端科学技術大学院大学助教授．理学博士．属性文法，自然言語の基礎理論，計算量理論などの研究に従事．電子情報通信学会，日本ソフトウェア科学会，LA 各会員．



中田　育男（正会員）

1935年生．1958年東京大学理学部数学科卒業．1960年同大学院修士課程修了．1960～79年(株)日立製作所中央研究所，同システム開発研究所勤務．1979年4月より筑波大学電子・情報工学系教授．理学博士．プログラム言語，言語処理系，ソフトウェア工学などに興味を持っている．著書「コンパイラ」(産業図書)，「基礎 FORTRAN」(岩波書店)，ソフトウェア科学会，電子情報通信学会，ACM，IEEE 各会員．