1N-2　属性文法システムＣＡＧを利用したモンタギュー文法の実働化

堺　和宏　　田村直良
(東京工業大学　工学部)

1．はじめに

属性文法では、意味記述と構文規則とを一体として一つの文法で表し、プログラミング言語の意味記述を形式的に行なうことができる。その手法を自然言語処理に応用するための条件付属性文法評価システムがFranz Lisp上で実現され、英語文の構文解析が有効に進められることが確認されている [1]。本研究では、さらに意味解析を行なうことを目的とし、モンタギュー文法における英文から内包論理への翻訳を実働化したのでそれについて報告する。

2．ＣＡＧの概要

条件付属性文法（Conditioned Attribute Grammar,以下ＣＡＧと呼ぶ）とは、属性文法における各生成規則に適用条件を付加したものであり、一つの文法規則は次の形式で記述される。

（生成規則） when （適用条件）
with （属性定義式）

適用条件を与えることによって曖昧性の除去や解析時のチェックをすることができる。属性定義式は、属性間の関係を表す等式である。属性とは、文法記号が持つ情報を値として持つ変数であり、機能的に次の２つに分類される（図１参照）。

(1) 導出木の根から葉の方向に情報を流す相続属性
(2) 導出木の葉から根の方向に情報を流す合成属性

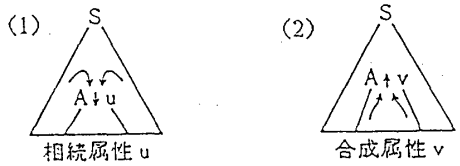

図１

我々の研究では、この属性定義式を用いることによって英文から内包論理表現への翻訳を実現している。

3．モンタギュー文法の概要

モンタギューは、全ての言語（自然言語、形式言語）について意味解釈までし得るような普遍的文法体系を「ＵＧ*」のなかで数学的に厳密に規定している。その文法体系の概略を図２に示す。基本的な立場としては、全ての言語Ｌは、曖昧化関係Ｒによって（意味的に）曖昧性のない言語ＤＬに対応づけられ、ＤＬについて意味解釈が為される（図２の点線で示される径路）、ということになる。　興味深いのは、"翻訳"という部分である。これは、ＤＬ間の翻訳ということであり、数学的に厳密な定義が与えられている。今、ＤＬについての意味論が定義しにくい場合、しっかりした意味論を持つＤＬ'に翻訳することにより、間接的にＬの意味解釈が為されるわけである（図２の実線で示される径路）。

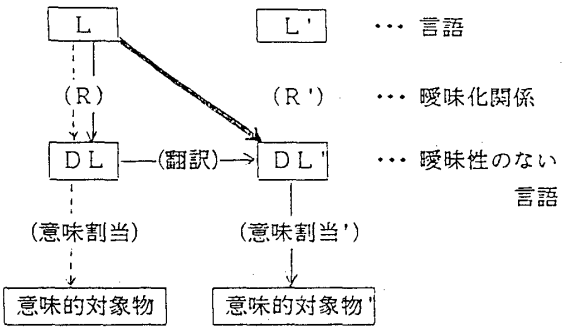

図２　ＵＧの枠組の概略

「ＰＴＱ*」のなかでは、Ｌとして英語(の断片)をとり、ＤＬ'として内包論理をとっている。英文から曖昧化関係Ｒに従って曖昧性のない形式である"分析樹"(ＤＬ)を作り、それを内包論理に翻訳する。　それに対し我々の研究では、Ｒというものを考えずに、英文の構文解析と同時に内包論理への翻訳操作を行なっている（図２の太線で示される径路）。

つまり、ＰＴＱの分析樹に対しては必ず内包論理表現が一通りに定まるのに対して、我々の方法では一つの構文解析木に対して複数の内包論理表現が与えられることもある。

*)ＵＧ，ＰＴＱはMontague.Rの論文。詳細は[2]を参照

4．英文から内包論理への翻訳

本節では、"John seeks a unicorn." という例を用いて英文を内包論理に翻訳する過程について説明する。

ＰＴＱではこの文に対して二つの分析樹が与えられる（図３－１，図３－２）。そして、それぞれの樹に対して内包論理表現は次のように与えられる。

Actualization of Montague Grammar on Attribute Grammar Evaluation System.
Kazuhiro SAKAI, Naoyoshi TAMURA
Tokyo Institute of Technology

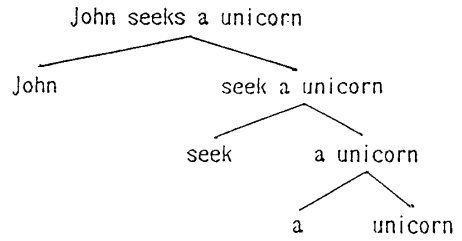

図３－１　非特定的解釈の分析樹

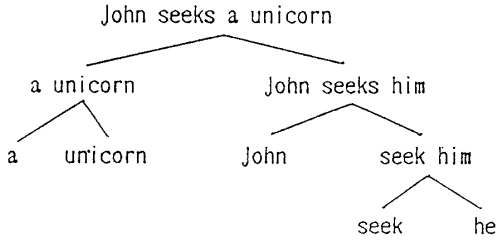

図３－２　特定的解釈の分析樹

非特定的：seek'(j, ^λQ ∃x[unicorn'(x) & [ˇQ](x)])
特定的　：∃x[unicorn'(x) & seek'(j, ^λP[ˇP](x)])

（ここで、j はJohnという個体、unicorn'は unicorn の翻訳、seek'はseek の翻訳を表す。 ^は内包化演算子、ˇは 外延化演算子を表す。）

両者の違いは、∃x[unicorn'(x)の位置である。つまり、非特定的な解釈は「ジョンが何らかの一角獣を探した」ということであり、特定的な解釈では「ある一角獣が存在し、ジョンはその一角獣を探した」ということになる。特定的な解釈の場合、名詞句である a unicorn は文の中で"量化"されているわけである。

ＣＡＧでは、同じ文に対して一つの導出木（図４）が与えられる。この木について二つの解釈を取り出す。

まず、非特定的な解釈については、辞書に各単語（終端記号）の合成属性生起としてＰＴＱと同様の翻訳を与えておき、それらを導出木に沿って属性評価することにより、ＰＴＱと同等の内包論理表現が得られる。これは、ＰＴＱでの方策をほとんどそのまま実現している。

一方、特定的な解釈の場合には、名詞句(a unicorn)を量化せねばならない。これは、まず名詞句の翻訳を一旦保管しておいてあとから保管庫から取りだす、という Cooperの"名詞句保管"の手法を用いて解決される（詳細は[2]他を参照）。我々の研究では先の非特定的な解釈の方法にこの名詞句保管の手法を取り入れ、同時に二つの翻訳結果を得られるようにする。

先の例では、名詞句(np)が次の生成規則で生成される。

```
np --> a, nomhd.
```

np, a, nomhd の合成属性生起 np↓.np, a↓.a, nomhd↓.nomhd はそれぞれ名詞句、冠詞、語幹の翻訳を値として持ち、属性定義式は次のように与えられる。

```
np↓ = (np_det_nom  a↓  nomhd↓)
```

np_det_nomは Lisp関数である。この定義式により np↓には、まず a↓,nomhd↓から合成される翻訳が与えられる。

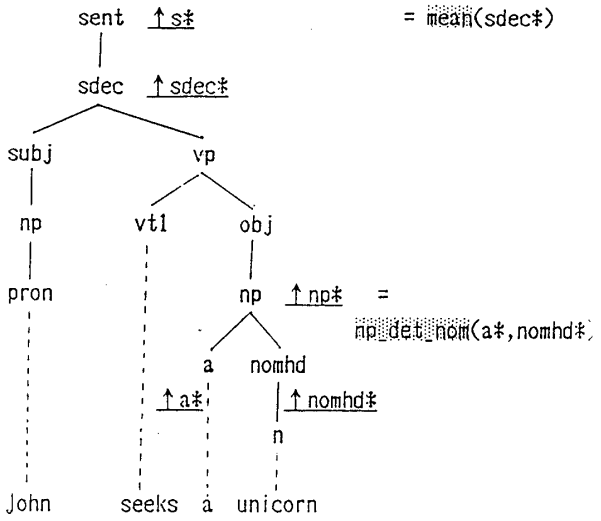

図４　ＣＡＧによる導出木

```
(20 (sent nil (s↓))
    =>
    ((sdec nil (form s-psn s-nbr sdec↓)))
    when
    t
    with
    ((s↓_0 = (mean sdec↓_1))))
```

図５　"文(sent) --> 平叙文(sdec)"の規則
(form,s-psn,snbr は構文解析用の属性,
sdec↓,s↓の値はsdec,sentの翻訳)

次に、その翻訳に名前(X)をつけて名詞句保管のためのリストに保管し、np↓にはXを使って作った翻訳を付け加える。以上、二つの翻訳を要素とするリストが np↓の値として与えられる。文(sent)が生成された時、meanというLisp関数で属性評価をすると、前者から非特定的解釈の翻訳が得られ、後者からは保管されていた翻訳をXに返すことによって特定的解釈の翻訳が得られる。
（図５に sent --> sdec の文法記述例を示す）

５．おわりに

条件付属性文法ＣＡＧによって、意味解析の記述をすることを目的としてモンタギュー文法を取り上げ、英文から内包論理表現への翻訳を実働化した。それによってＣＡＧの記述性の高さを確認した。現在、主部－述部，他動詞－名詞句（－名詞句），be動詞，受動文について内包論理への翻訳ができる。文法規則数は３４，辞書項目は約８０である。今後は、意味表現形式として他の形式（フレーム理論に基づくものなど）の記述を考えたい。

[参考文献]
[1]田村,高倉,片山：自然言語処理を目的とした属性文法評価システム，ｺﾝﾋﾟｭｰﾀｿﾌﾄｳｪｱ Vol.3 No.3, 1986.
[2]白井賢一郎：形式意味論入門，産業図書，1985.