

TR-110

ユニフィケーションによる
意味解析

安川秀樹

May, 1985

©1985, ICOT

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03) 456-3191~5
Telex ICOT J32964

Institute for New Generation Computer Technology

ユニフィケーションによる意味解析

安川秀樹（ICOT研究所 第三研究室）

我々は文脈理解のための計算機上でのモデルとして小学校の国語の教科書中の文章を対象とした質問応答システムDUALSを開発した。この論文ではDUALSの意味解析モジュールについてのべる。

ここでいう意味解析とは、入力された文の意味を表現する記述を求めることを指す。すなわち、文の表す意味の把握、それを表現するための基本的な手法、並びに句構造に従った文の意味の合成という三点について論じる。

本論文では、状況意味論[Barwise, Perry 83]を基に文の意味の解析を行っており、意味の表現形式及び、意味合成の基本的な手法として状況意味論における基本的な要素であるタイプ・システムを導入している。意味合成のための基本的な機構は、文法記述と同様にユニフィケーションであり、文法記述言語を拡張することにより、状況意味論のタイプ・システムに関するユニフィケーションを導入し、構成的に意味解析を行なっている。また、状況意味論のタイプ・システムは、おなじくDUALSの談話解析のために開発された状況意味論に基づくプログラミング言語CIL[Mukai 85]の基本要素となっている。

1章では意味解析手法の概要についてのべる。

2章では代表的な語彙についてその意味と表現についてのべる。

3章では構成的意味(Compositional Semantics)に基づく句の意味の合成についてのべる。

最後に4章では、現在の意味解析の問題点について述べる。

1. 意味解析手法の概要

(1) 文の意味とその表現

意味解析とは、文の意味をあらわす表現を求めることといえることができる。したがって、ここでは、まず文の意味をどうとらえるか、また、それをどのように表現するかということについて考察する。ここでの議論は状況意味論に基づくものである。

状況意味論では、文の意味を、『ある文の発話された状況と、その文の発話によって記述された状況の関係である』と定義している。すなわち、単純には、次の式で表現される。

$$1) \quad d, c [\phi] e$$

上式中、 d は談話状況(Discourse Situation)であり、文が発話された場所・時・話者・聞き手などの情報を含む。また、 c は結合状況(Connective Situation)であり、話者がどのような表現で何を言っているかという情報をふくむ。この d と c が文が発話された状況を形成する。一方、 e は、文の発話が、 d, c の下に記述する状況であり、記述状況(Described Situation)とよばれる。上式は、文 ϕ の発話の意味が、 d, c と e の関係 $[\phi]$ として表されることを示している。

したがって、文の意味の表現は、状況 e と、文の発話によって決まる d, c, e に関する条件を記述することが必要である。ここでは、状況意味論の基本要素であるタイプを意味の表現形式として用いる。

タイプ・システムの詳細についての説明は文献[Barwise 84], [Mukai 85]に譲り、以下に簡単な説明をあたえるにとどめる。

一般にタイプは、不定項(indeterminate)と条件からなり、次のように表わされる。

$$2) \quad [x | C]$$

2)の x は不定項、 C は条件である。

不定項は、あるタイプの実体(インスタンス)であり、individual, location, situationなどの基本タイプ、もしくは、2)のように定義された複合タイプ(complex type)の実体を示す。

条件は、3)に示すような時空領域を伴った事実(located fact)の集合であり、不定項 si に $t \cdot s$ に関連する制約条件を記述する。

3) {in(sit*S, loc*L, (speaking, ind*A), 1)}

不定項は、一般に次のように表わされる。

4) タイプ名*Prolog変数

Prolog変数は、タイプの実体を識別するための識別子として用いられる。たとえばsit*Sは、situationのタイプの不定項Sit、すなわち、ある状況を表わす。

図中#は、あるタイプの定数を表わす。たとえばind#rollsanは、individualのタイプの定数rollsan、すなわちロールさんという個人を表わす。定数は、不定項の特殊な場合であり、タイプの実体が単一である場合に対応する。たとえば、ロールさんのような個人や太平洋のような唯一の存在に対応する。定数は次のように表わされる。

5) タイプ名#アトム

定数は、アトムで示されるタイプの定数を記述する。

3)は、ある時空領域(spacio-temporal location)loc*Lにおいて、個体(individual)ind*Aが話しているという条件を、状況sit*Sが含むということを示している。sit*Sは、簡単に言うと、「誰かが話している」という状況を示すタイプである。

ここで、不定項loc*Lとind*Aは、それぞれ、「話している個体(人)」、「誰かが話しているロケーション」を示しており、3)で示した条件のパラメータと考えることができる。たとえば、loc*Lとind*Aにそれぞれlocとspeakerという名前を与えることにより、その名前を介して、タイプのパラメータにアクセスできる。

DUALSでは、このようなタイプを表わすのに、LFGの機能構造(F-Structure)を用いている。機能構造は、属性とその値とからなる構造的な表現である。たとえばと3)の条件およびパラメータloc*Lとind*Aにより定義される状況のタイプは次のように表わされる。

6) type	self	sit*S
	loc	loc*L
	speaker	ind*A
cond		{in(sit*S, loc*L, (speaking, ind*A), 1)}

すなわち、タイプは属性typeの値として表され、タイプの不定項をその属性self、条件

を属性condの値とし、各パラメータは、その名前を属性名とする属性の値として表わされる。各語彙項目、文法規則中の各非終端記号などの構文要素は、その意味記述をこのような形式で機能構造中に持っている。

(2) ユニフィケーション

意味解析の手法は、構成的意味論(compositional semantics) に従ったもので、句構造に従い、部分の意味から全体の意味を合成していく。

すなわち、文法規則の子ノードの持つタイプの表現から親ノードのタイプの表現を合成していく。その際に用いる基本的なメカニズムは、構文解析の場合と同様ユニフィケーションである。

ここで、簡単に機能構造のユニフィケーションについて説明する。

機能構造 f は、以下のように定義される。

$$\begin{aligned} Sy &= \{ sy \mid sy \text{ はシンボル} \} \\ T &= \{ t \mid t \text{ は項 (ターム)} \} \\ F &= \{ f \mid f \text{ は機能構造} \} \\ S &= \{ s \mid s = \{ x \mid x \in Sy \vee x \in T \vee x \in F \} \} \\ \\ f &= sy \ (\in Sy) \text{ あるいは} \\ f &= t \ (\in T) \text{ あるいは} \\ f &= s \ (\in S) \text{ あるいは} \\ f &= \langle l, v \rangle \mid l \in Sy \wedge v \in F \end{aligned}$$

また、機能構造 f の上の変数も存在する。

機能構造 f_1 と機能構造 f_2 のユニフィケーション $f = \text{unify}(f_1, f_2)$ は、以下のように定義される。 $p\text{-unify}(x, y)$ は、導出原理 (Resolution Principle) おけるユニフィケーションを示す。

case1: $f_1 = f_2$ ならば、 $f = f_1(f_2)$ 。

case2: $f_1, f_2 \in T$ の場合、 $f = p\text{-unify}(f_1, f_2)$ 。

case2: f_1 と f_2 の少なくとも一方が機能構造変数の場合、例えば、 f_1 が機能構造変数とすると、 f_2 を f_1 に代入し、 $f = f_2$ 。

case3: $f_1, f_2 \in S$ の場合、 $f = f_1 \cup f_2$ 。

case4: $\langle l, v_1 \rangle \in f_1 \wedge \langle l, v_2 \rangle \in f_2$ なる全ての l に対して、 $\langle l, \text{unify}(v_1, v_2) \rangle \in f$ かつ、 $\langle l, v \rangle \in (f_1 \cup f_2) - (f_1 \cap f_2)$ なる全ての l に対して、 $\langle l, v \rangle \in f$ 。

例えば、

```

7) unify([a 1], [a 1]) = [a 1]
   unify( a 1 , a 1 ) =  a 1
                        b 2  c 3      b 2
                                c 3

```

となる。また、属性値が相反する場合には、ユニフィケーションは、失敗する。機能構造間のユニフィケーションは、以上のように定義されるが、タイプを表す機能構造のユニフィケーションに関しては、拡張が必要である。

タイプを表す機能構造 t は6)に示したように、その不定項を self 属性の値として持ち、パラメータをその名前の属性の値として持つ。タイプの条件は、 cond 属性の値として、集合により表される。不定項は、先に示したように、Prologの項によって表される。問題は、タイプ間の両立性(compatibility)と条件の併合(merge)である。以下にタイプ間のユニフィケーションの定義を示す。

タイプを表す2つの機能構造 t_1 と t_2 のユニフィケーションを $t = \text{unify}(t_1, t_2)$ とする。

case1: $\langle \text{self}, x_1 \rangle \in t_1, \langle \text{self}, x_2 \rangle \in t_2$ なる x_1 と x_2 が同じタイプの不定項の場合。
 x_1 と x_2 のどちらか一方(例えば、 x_1) が定数不定項であれば、 $\langle \text{self}, x_1 \rangle \in t$ 、かつ、 $\langle \text{cond}, c_1 \rangle \in t_1, \langle \text{cond}, c_2 \rangle \in t_2$ なる c_1 と c_2 に対し、 $\langle \text{cond}, c_1 \cup c_2 \rangle \in t$ 。

x_1 と x_2 が共に定数不定項でない場合、 $\langle \text{self}, p\text{-unify}(x_1, x_2) \rangle \in t$ 、かつ、 $\langle \text{cond}, c_1 \cup c_2 \rangle \in t$ 。

$\langle l, v_1 \rangle \in t_1 \wedge \langle l, v_2 \rangle \in t_2 \wedge l \neq \text{self}, \text{cond}$ なる全ての l に対して、 $\langle l, \text{unify}(v_1, v_2) \rangle \in t$ 、かつ、 $\langle l, v \rangle \in (t_1 \cup t_2) - (t_1 \cap t_2)$ なる全ての l に対して、 $\langle l, v \rangle \in t$ 。

case2: $\langle \text{self}, x_1 \rangle \in t_1, \langle \text{self}, x_2 \rangle \in t_2$ なる x_1 と x_2 が異なるタイプの不定項の場合。 x_1 と x_2 が基本タイプの場合、ユニフィケーションは失敗。

x_1 と x_2 のどちらか一方が複合タイプの場合、 $\langle \text{cond}, c_1 \rangle \in t_1, \langle \text{cond}, c_2 \rangle \in t_2$ なる c_1 と c_2 に対し、 $\langle \text{cond}, c_1 \cup c_2 \cup (\text{in}(\text{ds}, \text{ld}, (\text{same}, x_1, x_2), 1)) \rangle \in t$ 。

$\langle l, v_1 \rangle \in t_1 \wedge \langle l, v_2 \rangle \in t_2 \wedge l \neq \text{self}, \text{cond}$ なる全ての l に対して、 $\langle l, \text{unify}(v_1, v_2) \rangle \in t$ 、かつ、 $\langle l, v \rangle \in (t_1 \cup t_2) - (t_1 \cap t_2)$ なる全ての l に対して、

$\langle l, v \rangle \in t$ 。

上のユニフィケーションの定義のcase2 の2番目の条件は、たとえば、「機長のロールさん」という同格用法の名詞句の解析において、機長である人とロールさんという個人が同一であることを記述する場合、すなわち、機長という複合タイプの不定項とロールさんという複合タイプの定数をユニファイしようとした場合に生じる。それぞれのタイプは基本的には、個体(individual)のタイプなのだが、不定項は、異なるタイプ名を持つ。すなわち、各々

8) captain*V1, ind#rollsan

という不定項で定義される。この場合、機長という複合タイプの定義をたどっていけば、(シソーラス的な階層構造による知識ベースを仮定して)、最終的には機長の不定項もind*V1という形に縮約されるはずなのだが、意味解析ではそのような縮約は行わない。従って、実際にユニフィケーションを行わずに、ds(談話状況)に、(same, captain*V1, ind#rollsan) という形で、ユニフィケーションの情報を蓄わえておくだけとする。実際には、この両者のユニフィケーションは、談話構造解析において評価されることになる。

たとえば、“ロールさんがはっとする”という文の意味を合成する際、PP“ロールさんが”とV“はっとする”の意味が、各々9)と10)のように与えられているとする。

```
9) pcase ga
   case nom
   type self ind#rollsan
      cond {in(cs, ld, (refers, ad, ind#rollsan, rollsan, name), pol#1)
            in(ds, ld, (otoko, ind#rollsan), pol#1)}
```

```
10) type self sit*Sit
     loc type loc*Loc
     cond {in(sit*Sit, loc*Loc, (hattosuru, ind*Ind), pol#1)
           in(w, lu, (t-overlap, loc*Loc, ld), pol#1)}
```

```
subj type self ind#Ind
```

簡単のため、ここでの説明に関係のない属性は取り除いてある。

9)と10) の2つのタイプ表現と、次に示す文法規則により、文の意味が合成される。

```

11)  s(s(Pp,V),lds) =>
      pp(Pp,ldpp) : (pcase,ldpp) = ga,
                    (case,ldpp) = nom,
                    (subj,lds) = ldpp &
      v(V,ldv) : lds = ldv.

```

11) の文法規則中、lds, ldpp, ldvは、それぞれノードs, pp, vの持つ機能構造を指示する識別子である。また、(pcase, ldpp)等の表現は、機能構造中の属性値を参照するための記法であり、パス表現と呼ばれる。例えば、(pcase, ldpp)は、ppの持つ機能構造のpcase 属性の値を指す。ppに付記されたユニフィケーションの内、(pcase, ldpp) = ga と (case, ldpp) = nom の二つは、ppが文法機能subj（主語）として機能するための条件であり、構文的な制約条件である。

ユニフィケーション(subj, lds) = ldpp は、ppの意味表現（機能構造）を、s の意味表現のsubj属性の値とユニファイするということを示す。s のsubj属性の値は、[type ind*Ind]という機能構造であり、これと、9)に示されたppの機能構造とのユニフィケーションが行われる。また、V に付記されているユニフィケーションlds = ldv は、V の意味表現とS の意味表現をユニファイするということを示している。

この2つのユニフィケーションにより、12) に見られるように、hattosuru の引数がin d#rollsan にユニファイされる。すなわち、“ロールさんがはっとする”ことがユニフィケーションにより合成される。

```

12)  type  self  sit*Sit
      loc  type  self  loc*Loc
      cond {in(sit*Sit, loc*Loc, (hattosuru, ind#rollsan), pol#1)
            in(w, lu, (t-overlap, loc*Loc, ld), pol#1)}
subj  pcase ga
      case nom
      type self ind#rollsan
      cond {in(cs, ld,
                (refers, ad, ind#rollsan, rollsan, name), pol#1)
            in(ds, ld, (otoko, ind#rollsan), pol#1)}

```

12) が表していることは、rollsan という個人が、はっとしている状況であり、それが、すなわち、この文の意味である。

なお、9), 10), 12)の意味表現において、いくつかの特殊な記号や関係が現われていたので、ここで簡単に説明しておく。

表現中のcs、dsはそれぞれ結合状況と談話状況を示す。また、ldは、談話の位置 (discourse location)を表わし、その発話が起った時空的位置を示す。また、luは普遍的位罫 (universal location)、w は世界(world) を表わし、それぞれ最大の時空領域と最大の現実状況を示す。adは談話状況ldにおける話者を表わす。これらは、対象とする問題領域において既定、もしくは、ある文の発話について固定しているため、定数シンボルで表わしている。(ある文の解析においては定数と考えてよい。) 特殊な関係としては、refersとt-overlap がある。refersは、話者の参照に関する情報を表わすものである。

13) (refers, ad, 不定項、文字列、参照のタイプ)

13) がrefersの一般形であり、話者がある文字列(たとえばrollsan)によってある不定項を参照しているという参照関係を表わしている。参照のタイプは、照応関係を同定するために、DUALSのオブジェクト同定部で必要とする情報であり、その参照が、固有名詞によるものか、一般名詞によるものか、あるいは、代名詞やゼロ名詞によるものかといった情報を含む。

t-overlap は、2つの時空領域が、時間的に重なりを持つことを示す関係である。このような時空領域に関する制約条件は、すべてw という最大の状況の、luという最大の時空領域内で定義される。すなわち、時空領域は、すべての状況に対して普遍的なものとしている。

2 語彙の意味

ここでは、各種のカテゴリの語彙の意味について、概観する。語彙の意味は基本的にタイプで表わされる。

(1) 固有名詞の意味

固有名詞の意味は、タイプの定数となる。例えばロールさんは、前章の9)のような表現で表わされる。ここで重要なことは、タイプと行っても、その実体は唯一しか存在しないことである。これは、DUALSにおける一つの仮定と言えるが、固有名詞に対応するもの(タイプの実体)は、モデル上でユニークに定まるということである。また、固有名詞には、特にパラメータを設定していないので、固有名詞の意味は次のように表現される。

```
14)  type self {タイプ定数}
      cond {条件}
```

前章の9)に見られるように、条件には参照の関係(refers)が含まれている。固有名詞の場合、参照のタイプはnameとなる。

(2) 一般名詞の意味

一般名詞の意味は、不定項で表わされる。固有名詞の場合と異なり、一般名詞の意味を表わすタイプの実体は唯一ではない。従って、一般名詞の意味は次のように表現される。

```
15)  type self {不定項}
      cond {条件}
```

固有名詞の場合と同様、参照の関係が条件に含まれており、その参照のタイプはcommonである。

また、名詞類一般に関わる問題であるが、名詞の意味と時空領域との関係の問題がある。たとえば、犬という名詞の意味が、次のような条件によって定義されている場合を考える。

```
16)  type self ind*D
      cond {in(sit*S,loc*L,(inu,ind*D),pol#1)}
```

この定義によると、ある状況sit*S 中の時空領域loc*L において、inu という関係がな

りたっている個体 $ind * D$ が犬であるということになる。問題は、この場合の $sit * S$ と $loc * L$ は何かということである。次の文を考えてみればポイントはより明確になるだろう。

17) 私の妻は、小学生のころは走るのが速かった。

17) の文では、“私の妻”の意味を、16) のように、ある時空領域を伴ったものとする
と、“走るのが速かった”のは“私の妻”なのだが、少なくとも“走るのが速かった”時
は、“私の妻”ではない訳で、時空領域が構文・意味からは分からない。また、仮に、そ
の時空領域を id 、すなわち談話の時空領域と仮定しようとしても、すでに“私の妻”が死
亡していたりとか、すでに離婚していたりということも考えられるわけで、結局うまい説
明はない。

16) のように、時空領域を伴った状況により犬の意味を定義するのが適当かどうかとい
う問題もある。どちらかという、 $ind * D$ が犬であるという属性(property)を持っている
ということが、犬の意味と考えることもでき、そのこと自体は、時空領域($loc * L$) とは関
係がないということもできる。すなわち、時空領域をとみなわない事実として、名詞の意
味を定義することも考えられる。

現在の DUALS では、名詞の意味として与えられたタイプの中に出現する状況・時空
領域については、単純に、その名詞を支配する述語の状況・時空領域と同一にしている。

また、もう一つの問題として、一般名詞の用法の問題がある。すなわち、一般名詞がタ
イプで表わされるとして、それが、ある特定の实体を指すのか、あるいは、実体の集合を
指すのか、あるいは、そのタイプで表わされる概念全般を指すのかということである。

18) 飛行機が、とんでいる。

19) 飛行機が、並んでいる。

20) 飛行機は、とぶ。

18) は、ある特定の飛行機、19) は飛行機の集合、20) は、飛行機一般を指すと考えら
れる。(18)にも集合的解析はあるが。)

これらは、日本語の名詞の単複同型である点、あるいは限定表現が省略可能である点な
どの構文的な要件の他に、動詞の意味にからむ問題、語用などの問題に起因する。これら
については、3章で再考する。

(3) 代名詞の意味

代名詞のうち、人称代名詞“私”、と“あなた”などの1・2人称代名詞の意味は、談

話状況から決まる。すなわち、1人称代名詞は、談話状況における話者であり、2人称代名詞は、聴き手となる。このような取扱いによって、“私”や“あなた”という語の意味が、発話のおきた状況の下で自然に決めることができるだけでなく“私”や“あなた”という表現の効率 (efficiency) を自然に表わすことができる。

3人称代名詞、たとえば、“彼”や“彼女”は、結合状況中の特定の“男である個体”あるいは“女である個体”を参照するものとして考えられる。その参照関係自体は、オブジェクト固定部で明確にされる。意味解析のレベルでは、例えば“彼”に対しては、21) のような表現を与える。

```
21)  type self ind*K
      cond {in(cs,ld,(refers,ad,ind*K,kare,pro),pol#1)
            in(cs,ld,(otoko,ind*K),pol#1)}
```

“これ”、“あれ”、“それ”などの指示代名詞は、それらの表現によって参照されるものについて、特に制約はなく、それが状況でも個体でもかまわない。したがって、“彼”の場合のように、個体の不定項により定義されることはない。また、参照のしかた、つまり、コソアドの情報が必要となる。

以上のことより、“これ”、“それ”などの不定項（たとえば“これ”に対して、kore という特殊なタイプを考える）によって表現することとした。“これ”に対する表現は以下になる。

```
22)  type self kore*K
      cond {in(cs,ld,(refers,ad,kore*K,kore,pro),pol#1)}
```

(4) 動詞の意味

動詞の意味は、基本的に一般名詞と同じような形で表現される。大きな違いは、一般名詞の場合、基本的に、個体を示すのに対し、動詞の場合は、個体もしくは個体間に関する関係を示す。したがって、名詞類においては時空領域の解釈が不明確であったのに対し、動詞の場合は、時空領域は、その意味にたいして本質的な役割を果たす。

すなわち、名詞の意味は、ある個体とそれが有する属性をしめし、不変のものではないにしても、時空的に変化するようなものではないのに対し、動詞の意味は、時空的に刻々と変化し得る、個体間の関係と考えられる。

また、名詞類の意味は、不定項と条件だけから表現されており、パラメータは持たなか

ったが、動詞の意味は一般に幾つかのパラメータを持つ。たとえば、関係名、時空領域、各種の文法機能、極性などがパラメータとなる。例えば、“着く”という動詞の意味は次のようになる。

```
23) type self sit*T
    rel type self rel#tuku (あるいは、単にtuku)
    loc type self loc*L
    pol type self pol*Pol
    ga type self ind*A
    ni type self ind*G
    cond {in(sit*T, loc*L, (tuku, ind*A, ind*G), pol*Pol)}
```

また、動詞の時制は、時制辞（時制をあらわす助動詞）によって決まる。
なお、形容詞の意味も動詞と全く同様に表わされる。

(5) 副詞の意味

副詞の意味は、未解決の問題であり、現在の定式化は再検討の余地が多い。一般に副詞は動詞・文主要部などにかかり、その意味を修飾すると考えられる。メタな働きを持つと言っても良い。副詞に修飾された結果、動詞の意味に関して、話者の主観的な要素が導入される。

24) ふちがみさんはにっこりとほほえんだ。

25) もう準備は済んだ。

24) や25) の下線部が副詞だが、それぞれの意味は、極めて主観的なもので、語用的に理解されるものと言える。DUALSでは、副詞の意味としては、タイプを与えていない。副詞は、あくまで動詞を修飾するもので、動詞の意味であるタイプの条件を更に付加えるものとして扱っている。例えば、“にっこりと”の意味は、26) のようになる。

```
26) cond {in(sit*S, loc*L, (type-of, sit*S, nikkorito), pol#1)}
```

26) が示していることは、副詞は、動詞と結びつき、動詞の意味記述の不定項（type属性の値）sit*S とパラメータloc の値loc*L をに関する条件を付加えるということである。

type-of というのは、動詞の意味である不定項sit*S がnikkorito というタイプであるということ、すなわち、24) の例文でいえば、“ふちがみさんがほほえんだ”という状況が、“にっこり”という表現がふさわしいものであるということを示している。

DUALSでは、このように、副詞の意味の解析ということはあまりしていない。単に、記述として残しているということもできる。

なお、DUALSでは、“急に”、“今”、“このまま”などのような、名詞等の副詞的用法も、簡単のため、副詞として語彙項目をたてている。

(6) 助動詞の意味

助動詞のうち、受身の“れる”、“られる”や使役の“せる”、“させる”などは、純粹に構文的なものとして扱っている。すなわち、受動化や使役化を表す表層のマーカーとして扱っている。

また、“です”、“ます”などのていねいを表わす助動詞も構文的に扱われ、話者のていねいさを表わす属性(polite)が機能構造中に付加され、意味解析では特に分析はしない。

時制を表わす助動詞“る”と“た”は、独自の文法カテゴリーを与えられ、意味的には、文の意味を表わすタイプの条件に、時制に関する情報を付け加える。すなわち、文に意味として得られたタイプで表される記述状況(described situation)が、談話状況と比較して、時間軸上でどのような位置にあるかという情報が付け加えられる。過去の場合は、記述状況の時空領域が談話の位置ldよりも時間的に先立っていることを示し、現在の場合はldと重複していることを示す。たとえば記述状況の時空領域をloc*Lとした場合、現在形ならば、

```
27) in(w, lu, (t-overlap, loc*L, ld), pol#1)
```

となり、過去形ならば、

```
28) in(w, lu, (t-precede, loc*L, ld), pol#1)
```

となる。

否定の助動詞“ない”は、動詞の意味の極性(pol 属性の値)を0にセットする。

たとえば、23)の“着く”の表現を参考にすると、“着かない”の表現は、29)のようになる。ただし、否定のスコープの問題は、残されている。

```
29) type self sit*S
      rel type self tuku
      loc type self loc*L
      pol type self pol#0
      cond {in(sit*S, loc*L, (tuku, ind*A, ind*G), pol#1)}
```

以上に比べ、動作や状態に対する話者の心的態度を表わす法の助動詞の意味は複雑で、十分な分析はできていない。

たとえば推量の“う”、“だろう”などが法の助動詞である。

法の助動詞は、独自にカテゴリを与えられ、文主要部と結びつく。

30) あと1時間で飛行機はマニラに着くだろう。

30) の文の意味は、発話の時点での状況における飛行機が、あと1時間飛行を続ければ、マニラに着くということであると考えられる。

すなわち、発話の時点の状況における飛行機に関するという条件付きの下で、あと1時間飛行を続けるという状況の意味のあるオプション(meaningful option)として、マニラに着くという状況が存在するという条件付き制約(conditional constraint)が、30) の意味と考えられる。このように、法に助動詞の意味は、一般に条件付き制約と考えられ、31) のように表わされる。

```
31)  type self sit*S
      loc  loc*L
      cond {moc(sit*S1,sit*S,sit*D,will*W))}
```

31) のsit*S が、制約の後件、すなわち意味のあるオプションであり、sit*S1が制約の前件、sit*D が制約の条件である。30) の文でいえば、sit*D が発話の時点の飛行機の関する状況、sit*S1があと1時間飛行を続ける状況、sit*S がマニラに着くという状況である。法の助動詞は、文主要部をとり、文となり、タイプのパラメータを介して、制約の条件、前件、後件を受けとり、31) の表現を完成する。しかし、制約の条件は抽出するのが難しく、現在は、正しく取り扱い得ていない。なお“もし～ならば…する”という条件文の意味も、法の助動詞と同様に、条件付き制約として表される。

3 句の意味

ここでは、文法規則に従って、句の意味がどのように合成されるかについて、代表的な例を示す。

(1) 名詞句の意味

固有名詞、代名詞、一般名詞だけからなる名詞句の意味は、それぞれの語彙の意味と同じであり、単純に、名詞句の機能構造と各語彙の機能構造をユニファイすることにより得られる。以下に、そのルールを示す。(これ以降で例示する文法規則中の関数表記は、意味合成に関するものだけで、構文的関係だけを示すものは取り除いてある。)

32) $np(np(Stname), Np) \Rightarrow name(Stname, Name) : Np = Name.$

33) $np(np(Stpro), Np) \Rightarrow pro(Stpro, Pro) : Np = Pro.$

34) $np(np(Stnom), Np) \Rightarrow nom(Stnom, Nom) : Np = Nom.$

関係節化された名詞句については、主名詞が埋め込み文の格を支配するか否かという2つの場合がある。

主名詞が格を支配する場合は、次の35)の規則で意味が合成される。

35) $nom(nom(Sts, Stnp), Nom) \Rightarrow$
 $s(Sts, S) : (cond, Nom) = (cond, S) \ \&$
 $nom(Stnom, Nom1) : Nom = Nom1, Nom1 = controller(pp).$

埋め込み文中の省略された格と主名詞の関係は、構文的なコントロールに関するユニフィケーションで行なわれ、埋め込み文の意味の持つ条件と名詞類(nom)の持つ条件の和が取られる。

一方、埋め込み文と主名詞の間に格関係がない場合、すなわち、埋め込み文の意味と主名詞の意味の間の関係が文脈的・語用的に決まる場合は、問題があり、現在は、両者の間に何らかの関係があるということを記述する条件をたて、両者の条件の和に付加している。

36) $in(cs, Id, (rel*R, ind*I, sit*S), pol*1)$

すなわち、文脈的・語用的に決まる、埋め込み文の意味と主名詞の意味の間の関係を、関係のタイプの不定項rel*Rを用いて表現している。

また、名詞句の用法に関する問題として、その意味が何を指すかという参照に関連するものがある。

37) いい年をして赤い服を着ている奴はおかしい。

37) の文が発話された状況で、話者の資源状況中に、‘太郎はもう十分いい年で、かつ赤い服を着ている’という状況、すなわち、37) の関係節化された名詞句の意味が参照している個体がすでに存在すれば、発話の意味としては、‘太郎はおかしい’ということが対応する。これは、文脈参照的用法(referential use)と呼ばれる。

一方、話者の資源状況中に、対応する個体が存在しない場合は、‘いい年で赤い服を着ている’という属性を持った個体(誰かは分らないが)を参照することになる。これは属性的用法(attribute use)と呼ばれる。属性的用法の場合は、関係節化された名詞句の意味が、話者の資源状況に抽入されることになる。資源状況は、結合状況に含まれるのであるが、属性的用法の場合は、これまでに述べたような単純な参照関係を結合状況に付け加えるだけではだめであることは明らかで、結合状況に、関係節化された名詞句の意味自体を付け加えねばならない。

現在の意味解析では、全て、文脈参照的用法としての表現を合成している。属性的用法か否かの区別は、これまでの文脈に対応するものが存在するか否かのチェックで行っており、これは、オブジェクト同定部で処理され、必要に応じて、資源状況に属性的用法に対応する名詞句の意味を注入している。

同格名詞句については、主名詞と埋め込みの間に語用的な関係がある関係節化名詞句と同じ扱いをする。ただし、

38) ～との知らせ。

39) ～と知らせる。

のように、同格名詞句に対応する文がある場合は、主名詞(38)の知らせ)が動詞と同じ意味を持っており、対応する文と同じ意味を構成する。その意味が、同格名詞句の意味となる。その場合の意味合成規則は40)のようになる。

40) $np(np(Sts, Stn1), Nd) \Rightarrow$
 $s(Sts, S) : (xcomp, Np) = S, (cond, Np) = (cond, S) \&$
 $n(Stn1, N1) : Np = N1, (sahen, N1) = c +.$

この場合は、いわゆる派生名詞の取り扱いと同じであり、対応する動詞に関する格支配関係は、文法関係を定めるユニフィケーションで行われる。すなわち、前掲の9), 10), 12)と同様に文法機能をキーにユニフィケーションを行うことで埋め込み文と主名詞に対応する動詞の意味的な関係が抽出される。それに加えて、埋め込み文の条件と主名詞条件の和

がとられ、同格名詞句の条件となる。したがって、38) の意味は、‘～と知らせたこと’
といった表現になり、名詞化構造の場合と同様になる。

名詞化構造の意味は、埋め込み文の意味そのものとなる。

限定辞を含む名詞句の意味は、難しい問題であり、より詳細な解析が必要である。たとえば、格助詞“の”を伴う名詞句が、限定辞として名詞類にかかる場合が考えられる。

41) 機長のロールさん

42) ふちがみさんの顔

43) 50人の乗客

41) は、状況意味論でいう同格用法(appositive use)に対応し、機長で示されるタイプとロールさんで示されるタイプがおなじもの、すなわち、機長である人（個体）がロールさんという人（個体）とおなじものであるということを意味する。

42) は、所有（所屬）関係を表す。つまり、ふちがみさんという個体の一部としての顔という特定の实体を指す。

43) は、数量を限定する表現で、乗客で示されるタイプの实体の集合で要素数が50であるものを示す。

これらの用法の区別は、限定辞中の名詞句と、被限定名詞句の意味的な共起関係によって決定される。しかし、構文・意味解析中に、全ての共起関係のチェックを行なうのは難しい（少なくとも現在の枠組みでは）。従って、考えられる方法は2つ残されている。1つは、両者の共起による用法の区別は、後の談話構造解析にまかせ、意味解析では、2つの名詞句が、“の”という格助詞による限定表現で共起したという記述のみを残しておく方法である。例えば、44) のような記述を残すことである。

44) in(sit*S, loc*L, (co-occur, no, kityou*K, ind#rollsan), pol#1)

もう1つの方法は、文の表層レベルでそれぞれの用法を区別してしまうもので、同格用法に“の1”、所有用法に“の2”といったように、異なる種類の単語（表層）を割当てて、欠く用法の意味を意味解析中に抽出してしまう方法である。例えば、同格用法の場合、次の45) と46) で、その意味が抽出される。

45) np(np(Stdet, Stnom), Np) =>
det(Stdet, Det) : (spec, Det) =c no1, (type, Np) = (type, Det) &
nom(Stnom, Nom) : Np = Nom.

```

46) det(det(Stnp, p(no)), Det) =>
    np(Stnp, Np) : Det = Np &
    [no1] : (spec, Det) = no1.

```

同格用法の“の”に“no1 ”という表層と、限定情報（spec属性の値）“no1 ”を与え、45) の規則で、限定辞中の名詞句と被限定名詞句の意味をユニファイすることにより、同格用法の意味を合成できる。限定表現は意味解析の研究として重要なものであり、欠くことのできないものであるという判断に基づき、DUALSでは、この方法を採用している。

一方、所有関係的用法は、47), 48) により解析される。

```

47) np(np(Stdet, Stnom), Np) =>
    det(Stdet, Det) :
        (spec, Det) =c no2 ,
        in(ds, ld, (part-of, (type, Np), (type, Det))<pol#1) < (cond, Np),
        (cond, Np) = (cond, Det) &
        nom(Stnom, Nom) : Np = Nom.

48) det(det(Stnp, p(no)), Det) =>
    np(Stnp, Np) : Det = Np &
    [no2] : (spec, Det) = no2.

```

また、いわゆる限量子、例えば“ある”、“その”、“この”等の限定限量子(Definite determiner) や、“全ての”、“多くの”等の一般限量子(General determiner)の解析は、意味解析の重要なテーマであるが、現時点では、その一部のみ（ある、その）を扱っている。

“ある”は名詞で示されるタイプの実体のある物を指し、“その”等の場合は、話者の資源状況に、そのタイプの実体がただひとつしか存在せず、その唯一の実体を指す。たとえば、“ある人”と“その人”は、それぞれ以下のように表される。

```

49) in(cs, ld, (type-of, ind*I, hito*H), 1)
50) in(cs, ld, (unique, ind*I, hito*H), 1)

```

また、一般限量子は、一般化された限量子(Generalized Quantifier)の考え方をを用いて解

析可能である。一般化された限量子の理論においては、名詞も動詞と同様に関係(Relation)をあらわすものとしてあつかわれ、限量子は、関係間の制約条件をあらわすものとして定式化される。たとえば、「すべての人は夢をみる」という文に対して、次の表現があらえられる。

51) in(sit*S, loc*L, (every, hito*H, sit*D), 1)

52) in(sit*D, loc*L, (dreaming, ind*I), 1)

53) in(hito*H, loc*L, (hito, ind*I), 1)

上式中、関係every は、状況間の制約を表し、hito*Hが成り立つならばsit*D も成り立つということを表す。すなわち、人ならば、夢をみるという事を表す。現在、一般化された限量子を用いて、限定表現の意味を合成する方法を検討中である。

(2) 文主要部の意味

文主要部の意味は、それによって記述される状況のタイプである。すなわち、状況のタイプの不定項とその条件からなるタイプによって表される。また、各文法機能、時空領域、極性(Polarity)等をパラメータとして持つ。例えば、“ロールさんが、はっとする。”という文主要部(文ともなる)の意味は、次のように表現される(12)の再掲)。

```
54) type self sit*Sit
      loc type self loc*Loc
      cond {in(sit*Sit, loc*Loc, (hattosuru, ind#rollsan), pol#1)
            in(w, lu, (t-overlap, loc*Loc, ld), pol#1)}
```

subj pcase ga

```
      case nom
      type self ind#rollsan
      cond {in(cs, ld,
                (refers, ad, ind#rollsan, rollsan, name), pol#1)
            in(ds, ld, (otoko, ind#rollsan), pol#1)}
```

文主要部の意味の基本的な部分は述語の意味から決まり、述語にかかっている要素(ppやadv)は、述語のパラメータとユニファイされたり、あるいは、条件を付加したりという形で、述語の意味を具体化したり修飾したりする。

以下に簡単に、文主要部の意味の合成の過程を、文法規則をたどりながら説明する。
 簡単のため、文法規則には、意味合成規則のみを付記してある。
 まず、説明に入る前に、一般的な形で示した文主要部の生成規則を示す。

$$55) \text{ snuc} \rightarrow X^* \ \& \ \text{vp} \\
X^* = \text{pp} \ \text{or} \ \text{adv} \ \text{or} \ \text{scomp}$$

55) 中、vpはいわゆる述語であり、格要素 (pp) や副詞 (adv)、あるいは補文 (scomp) の任意個の並びが、それに先行している。実際の文法規則では、カテゴリ変項Xやクリーン表現^{*}が使えないため、56) を展開した形になっている。

$$56) \text{ snuc}(\text{snuc}(\text{Stvp}), \text{Snuc}) \Rightarrow \text{vp}(\text{Stvp}, \text{Vp}) : \text{Snuc} = \text{Vp}.$$

56) は、述語だけからなる場合で、述語の持つ意味がそのまま文主要部の意味になる。

$$57) \text{ snuc}(\text{snuc}(\text{Stpp}), \text{Stsnuc1}), \text{Snuc}) \Rightarrow \\
\text{pp}(\text{Stpp}, \text{Pp}) : ((\text{pcase}, \text{Pp}), \text{Snuc}) = \text{Pp} \ \& \\
\text{snuc}(\text{Stsnuc1}, \text{Snuc1}) : \text{Snuc} = \text{Snuc1}.$$

$$58) \text{ snuc}(\text{snuc}(\text{Stpp}), \text{Stsnuc1}), \text{Snuc}) \Rightarrow \\
\text{pp}(\text{Stpp}, \text{Pp}) : \\
(\text{cond}, \text{Snuc}) = (\text{cond}, \text{Pp}), \\
(((\text{pcase}, \text{Pp}), (\text{type}, \text{Pp}), (\text{type}, \text{Snuc})), \text{pol}\#1) < (\text{cond}, \text{Snuc}) \ \& \\
\text{snuc}(\text{Stsnuc1}, \text{Snuc1}) : \text{Snuc} = \text{Snuc1}.$$

$$59) \text{ snuc}(\text{snuc}(\text{Stadv}), \text{Stsnuc1}), \text{Snuc}) \Rightarrow \\
\text{adv}(\text{Stadv}, \text{Adv}) : \\
(\text{cond}, \text{Snuc}) = (\text{cond}, \text{Adv}), \\
(\text{loc}, \text{Snuc}) = (\text{loc}, \text{Adv}), \\
(\text{rel}, \text{Snuc}) = (\text{rel}, \text{Adv}), \\
(\text{type}, \text{Snuc}) = (\text{type}, \text{Adv}) \ \& \\
\text{snuc}(\text{Stsnuc1}, \text{Snuc1}) : \text{Snuc} = \text{Snuc1}.$$

$$60) \text{ snuc}(\text{snuc}(\text{Stscomp}), \text{Stsnuc1}), \text{Snuc}) \Rightarrow \\
\text{scomp}(\text{Stscomp}, \text{Scomp}) : \\
(\text{cond}, \text{Snuc}) = (\text{cond}, \text{Scomp}).$$

(xcomp, Snuc) = Scomp &
 snuc(Stsnuc1, Snuc1) : Snuc = Snuc1.

57) は格要素 (pp) が文主要部にかかっている場合である。pp は、助詞の情報を文法機能として、文主要部にユニファイされ、文法機能を橋渡しに、文主要部のタイプのパラメータと pp のタイプがユニファイされる。この様子は、前掲の 9), 10), 12) に具体例が見られる。なお、pp の意味は、pp の下の名詞句の意味となる。但し、助詞の情報が、構文的に付加わっている点異なる。

58) は、句構造的には、57) と同じであるが、pp が述語の格を支配していない場合、すなわち、pp が自由格要素である場合である。自由格要素は、文主要部の意味を修飾するものと考えられる。すなわち、自由格要素の意味と文主要部の意味が助詞で表される関係にあるということができる。ここでは、

61) (((pcase, Pp), (type, Pp), (type, Snuc)), pol#1)

という条件を付加えることで、それを表現している。(pcase, Pp) は助詞の情報、(type, Pp) と (type, Snuc) は、各々 pp と snuc の意味 (タイプ) の不定項であり、例えば、“飛行機で．．．に着く” に対して、

62) ((de, hikouki*H, tuku*K), pol#1)

という事実 (fact) を条件として付加する。“de” という関係は、それ自身は、助詞の表層を示すに過ぎないが、タイプ間の意味的な関係を示し、意味解析の後にその解釈が行なわれる。

59) は、文主要部に副詞がかかる場合である。副詞のパラメータと文主要部のパラメータがユニファイされ、両者の条件も和がとられる。条件としては、前述したように、文主要部の意味が、副詞で示される関係で修飾されるという情報が付加される。

62) は、補文をとる場合である。補文の意味は、状況のタイプであり、それは、xcomp という文法機能として、文主要部とユニファイされる。

(3) 文の意味

文の意味は、その文によって記述される記述状況を表すタイプとなる。文は一般的に書くと、次の規則で生成される。

$$63) s \rightarrow (sadv) \& snuc \& (t) \& (modal) \& (sp)$$

$sadv$ は文副詞、 $snuc$ は文主要部、 t は時制を表す助動詞、 $modal$ は法の助動詞、 sp は終助詞である。

文副詞は、文主要部における副詞の扱いと同じように、解析される。

時制を表す助動詞は、文の意味を表すタイプの条件に、談話の位置 ld と、その文のタイプの位置との関係を付加える。

また、終助詞は、文の形、例えば疑問文かどうかなど、を表し、機能構造中に、 s_type という属性の値として、その形を登録する。ちなみに、 s_type 属性の値は、平叙文の場合は $inform$ 、疑問文の場合は ask となる。

一番、問題となるのは、法の助動詞がある場合である。法の助動詞に意味の分析自体は、言語学・自然言語理解における大きな問題の一つであり、現在のDUALSでも、十分に納得できる解答は得られていない。ともあれ、法の助動詞の場合は、次の文法規則により意味解析を行なっている。

$$\begin{aligned} 64) s(s(Stsnuc, Stmodal), S) => \\ & snuc(Stsnuc, Snuc) : \\ & (xcomp, S) = Snuc, (cond, S) = (cond, Snuc) \& \\ & modal(Stmodal, Modal) : S = Modal. \end{aligned}$$

この場合、文の意味は、法の助動詞の意味によって決定され、文主要部の意味は、法の助動詞の意味のパラメータとしてユニファイされる。法の助動詞の意味は、前述のように、条件付き制約として表される。

4 意味解析の問題点と今後の課題

意味解析の問題点等については、これまでの各章において、折にふれ示してきた。もっとも基本的な問題としては、談話構造解析で用いられているプログラミング言語CILとのインターフェイスがあげられる。ここで示した意味解析では、機能構造とその上で定義された拡張ユニフィケーションを用いているが、機能構造中のタイプの表現とCILにおけるタイプの表現のインターフェイスが、統一のとれたものとなっていない。DUALSのシステム全体の一貫性という点から大きな問題であるといえる。CILにおけるユニフィケーションは、機能構造とそのユニフィケーションと極めて類似している。ある意味では、CILは、LFG等のユニフィケーション文法を、論理型言語の観点から整理し再構成したものとする事が可能であり、事実、CILは、ユニフィケーション文法を一般化した形で包含している。したがって、今後CILを用いて、意味解析を再構成することが、考えられる。その場合も、ここで述べたことは、そのまま適用できる。構文解析の効率等を考えると、CILをベースとするパーザの開発が必要となろう。また、文法意味解析のための枠組み（ユニフィケーション）を整理し、拡張することも重要な要素となろう。

意味解析自体の問題点としては、一般名詞の時空領域や、その様々な用法の解析の問題、代名詞やゼロ代名詞の参照の問題、副詞や法の助動詞の意味の解析に関する問題、一般限定詞の解析（スコープを含めて）等が挙げられる。これらの問題は、難しい問題であり、時間を要するテーマであるが、基本的な分析を積み上げてゆくことにより、少しずつ解決されてゆくものと思われる。これまでの検討から、状況意味論的アプローチが、これらの研究テーマを進めていく上での、有望なパラダイムであると確信している。

[参考文献]

- [Barwise 83] Barwise, J., Perry, J., "SITUATIONS AND ATTITUDES" , MIT-Press, Cambridge, Massachusetts, 1983.
- [Barwise 84] Barwise, J., "Lectures on Situation Semantics", Lecture Note of CSLI, winter quater, 1984.
- [Barwise 84] Barwise, J., Perry, J., "Shifting Situations and Shaken Attitudes" CSLI Report No. CSLI-84-13, 1984.
- [Cooper 84] Cooper, R., "Lectures on Preliminary Eliuss", Lecture Notes, 1984.
- [Cooper 84] Cooper, R., "Preliminary Eliuss", Lecture Notes , 1984.
- [Cooper 84] Cooper, R., "Fragments in Eliuss Style", Lecture Notes, 1984.
- [Gunji 84] Gunji, Y., "A PHRASE STRUCTURAL ANALYSIS OF THE JAPANESE LANGUAGE(draft)", 1984.
- [Mukai 85] Mukai, K., Horn Clause Logic with Parameterized Types for Situation Semantics Programming, to appear as ICOT TR-101, 1985.