OUKA
The University of Osaka
Institutional Knowledge Archive

Title	タイプ、メタファー、ポインター
Author(s)	杉本, 孝司
Citation	大阪外大英米研究. 1992, 18, p. 71-83
Version Type	VoR
URL	https://hdl.handle.net/11094/99154
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

https://ir.library.osaka-u.ac.jp/

The University of Osaka

タイプ，メタファー，ポインター

杉 本 孝 司

1　本稿[1]は杉本（1988）の延長線上に位置づけられ、メタファーが持つ形式意味論における意味公準的な役割をどのように論理体系（特にモデル）の中で反映させるべきかということに関する一試案を提供するものである。

2　Lakoff and Johnson（1980）や形式意味論の考え方に基づき杉本（1988）においてはメタファーの役割に関連して次のようなことを確認した。

（1）

1　メタファー概念は形式意味論における論理的意味公準と同じ役割をする
2　この意味公準は恣意的に決定されるものではない
3　それはNatural Logicにおける意味公準と考えられる
4　一般に意味公準は形式意味論における可能な解釈を規定する
5　Natural Logicにおいては可能な解釈とは「人間にとって可能な解釈」でなければならない
6　一般に解釈はモデル理論に基づく
7　従ってモデルは人間の概念形成を反映できるものでなくてはならない

結果次の5つは程度の差こそあれメタファー概念との関連においては[2]同じものであると考えられた。

（2）

杉　本　孝　司

1　意味公準

2　モデルに課された条件

3　可能な解釈に課された条件

4　我々人間の概念形成に課された条件

5　メタファー的概念

当然の結果としてこのような考え方を反映させ得る形式意味論的論理体系の構築が考えられる。Carlson (1977) やChierchia (1984) は部分的にではあるが、このような試みとして捉えられることも指摘した。前者においては個体と種とその顕現形態の関係が、後者においては個体とより高階の述語との関係が、それぞれモデル内におけるEntities間の関係として捉えられており、それは例えばそれぞれ次のようなメタファーの反映として考えられることを指摘した。

(3)

(Sugimoto (1988), p. 185 & p.188)

(29) DOGS ARE AN INDIVIDUAL
DOGS ARE AN ENTITY
DOGS ARE A KIND-LEVEL INDIVIDUAL
DOGS REFER TO A PARTICULAR KIND-INDIVIDUAL

(36) PROPERTIES ARE INDIVIDUALS
COMMON NOUNS ARE INDIVIDUALS
To-INFINITIVES ARE INIDIVIDUALS
GERUNDS ARE INDIVIDUALS

3　人間とはまさにおもしろい動物で同じものを異なった観点から違うものとして捉えることもできれば、違うものを同じ観点から同じものとして捉えることもできる．「ものの見方」の違いということになるが、これらが言語表現のレベルで確認できることは、例えば前者に関しては

(4)

An assembly of entities may be conceptualized, either in terms of its constituent members, i. e., as a multiplex, or as an undifferentiated mass. (Taylor (1989), p. 128)

などに見る通り同じ牛の集まりでも

(5) cows (*multiplex*)－cattle (*mass*) (Lakoff (1987), p. 428)

と呼び分けられる、などといった事実があげられるし、後者に関しては

(6)

a. There are guards posted along the road.

b. There is a fence along the road.

(Lakoff and Brugman (1986), p. 444)

と同じ “along the road” であっても(a)においては点的存在、(b)においては線的存在といった全く異なったものの在り方が表されている、といった例やLakoff and Johnson (1980) が示した数多くのメタファーの例などがあげられる。メタファーは多くの場合後者のように異質のものを理解するのに一つの視点あるいはものの見方を提供することによって共通認識の基盤を与えてくれると考えられるが

(7)

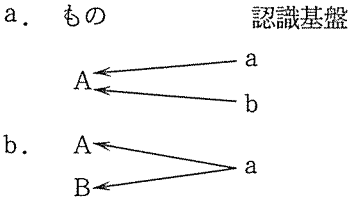
The essence of metaphor is understanding and experiencing one kind of thing in terms of another.

(Lakoff and Johnson (1980), p. 5)

同じ者を理解するのに異なった視点を与えていると解することのできる前者の場合も含めて、「理解および認識の基盤を与える」という意味において両者とも（概念化に関与する）メタファーと考えてもよいと思われる．簡単に

図式すれば[3)]

（8）



4　標準的タイプ理論（例えば　Dowty et al.（1981）などを参照）に従って意味論を整備すれば言語学的に見てやや不都合と思える現象がいくつか生じてくる。中でも「タイプの過剰生産」とでも呼べるような（言語学的には明らかに）不必要と思える範疇が過度に（というか理論的には無限に）許容され且つ要求されるようになる。

（9）

a. John is nice.

b. Being nice is a virtue.

c. Virtue is a quality.

Johnは個体を、being niceは個体の属性を表す言語表現としょう。とすれば確かに(a)は個体の属性を述定する文として了解できる。しかしこのことは(b)は属性の属性を述定する文として理解されねばならないことを意味している。何故なら主語Being niceは属性を表し、述語is a virtueはその主語の属性を述定しているのだから。そして更に(c)においてはそれが属性の属性の属性を述定する文として理解されねばならなくなる。つまりis nice, is a qualityの述語はタイプ理論に基づく限り、異なった範疇に属する述語ということになる。言語学的には単なるVPであるこのような表現が何故異なった範疇に所属せねばならないのか、ということが大きな問題点として浮き上がってくる。同じ問題は次のような形でも生じてくる。

(10)

a. John is nice

b. i. Being home is nice.

ii. To be home is nice.

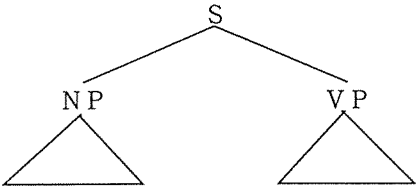
主語の位置は(a)が固体、(b)が属性を表す表現である。ということは(a)、(b)それぞれにおけるis niceという述語は二つのタイプ理論的には異なった範疇に属することになる. これも言語学的には大問題であると言える. もちろん自然言語の分析に無頓着な論理学者の中にはこのことが何故大問題なのか理解に苦しむといった風を装う人がいることも事実だが、われわれの言語直観も含めて言語表現のレベルでは明らかにそこに存在しない区別が（無限に）存在すると言っている訳だからそれ相当の根拠付けが要求されることは当然と言える。Chierchia（1984）が取り組んだのはまさにこの問題であって、このことは杉本（1988）でも簡単ではあるが触れた。（更に詳しくはChierchia（1984）及びその解説である白井（1991）第3章などを参照されたい。）

ここで見るタイプ理論の言語学的観点から見た場合のつまづきの原因は何かと考えれば、それは固体と属性を無反省的に且つ機械的に首尾一貫した方法で言語表現に適用していった結果である、と言うことができる。（9a）などの単純な場合を出発点にし、主語が固体を表せば、その主語の特徴を述べる述語は当然固体の属性を表すタイプとなり、今度はこの固体の属性を表す述語が（名詞化などによって）主語になれば、その主語の特徴を述べる述語は当然（主語の位置にある）述語の述語ということになり、述語のタイプが一階上がることになる。このプロセスが際限無く続く訳である。しかし例えば「述語の述語の述語の述語の述語」などといった述語が単純な主述関係を表す文の述語である、などといった状況は不自然としか言いようがない自然言語の在り方と言える. そのような意味的区別をシンタックスの上で表現している自然言語など存在しないであろう。ではどのようにすればモデル理論を組み込んだ形式意味論的枠組の中でラッセル・パラドックスのような論理的

矛盾を回避し且つ同時にタイプ理論が陥っている上に見たような自然言語の不自然な分析を乗り越えることが可能であるのか、ということが当然模索されることになってくる。[4]

5 (9)や(10)にまつわる問題は結局

(11)



において、NPの（意味的）範疇は恒に固体（タイプ理論的には e）、VPのそれは恒に属性（同 t/e ）[5]とする一般的な手段と根拠があれば解消されることになる．ここにおいて初めて(2)で見るようなメタファー概念とモデルに課された条件や可能な解釈とが（すくなくとも部分的には）等しい、という考え方が有効であるという可能性が見えてくる．それは高階のタイプを低階のタイプの視点から理解・認識するということによって生じる新しいものの捉え方であり、（8b）に見るメタファーの図式とも言える。つまり属性（being nice, virtue, being home, to be home, etc.）を固体とみなすメタファーと言うことができる。

(9), (10)においては(11)に見るごとく主・述関係においての問題を見た訳だが同様の問題は目的語と他動詞（そしてより一般的にはNPとその支配語句）の間にも見られる。

(12)

a. i. John knows Mary.

ii. John knows that Mary is pregnant.

b. i. John wants Mary.

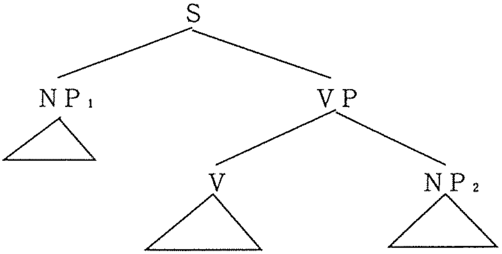
ii. John wants to visit Mary.

（12a）においては固体と命題、（12b）においては固体と属性がそれぞれ同一の動詞 know, wantの目的語として機能している。ここにおいてもタイプ理論上の要請から二種類の $know_1$, $know_2$二種類の $want_1$, $want_2$の設定をする、ということは容認し難い。(9), (10)に関して述べた問題の他に例えば

（13）　a. John wants Mary badly.
　　　　b. John wants to vist Mary badly.

に見るごとく、二つの$want_1$, $want_2$は二つの$badly_1$, $badly_2$を要求するなど、ことは深刻である。ここにおいても

（14）



NP_2の範疇は固体（タイプ理論的には e), Vのそれは固体から属性への他動詞（同（t/e）/e）とする一般的な方法があれば問題は解消することになる。高階のタイプを低階のタイプの視点から理解・認識するということ、つまり例えば命題（that Mary is pregnant）や属性（to visit Mary）を固体としてメタファー的に認識するという可能性を考えれば問題解決の糸口となることは明らかであろう。このような一つのメタファーを形式意味論のモデル内で反映させる糸口としてプログラミング言語であるC言語のポインターの考え方を次に見ることにする。

6　（15）は省略もあるが簡単なC言語のプログラムである。

(15)

```
#include 〈string.h〉
main
{
        char c;
        char *p;
        c='a';
        p=" personal computer";
        printf("%c", c);
        printf("%s¥n",p);
}
```

これをコンパイルして走らせればスクリーンに(16)の文字列が現れる。

(16)

```
a personal computer
```

C言語においては文字列用の変数がなく、一般に文字関係の変数は(15)のchar cにおけるcのように文字一つ（(15)では 'a'）を表す。そこで当然文字列の扱いをどうするかという問題が生じてくる。本稿との関係でC言語のおもしろい点もこの部分にある。文字配列という扱い方もあるが、それとは別に文字ポインター（以下単にポインター）というものが用意されている。(15)で宣言されている*pがそれでアステリスクを変数の前に置くことによって示される。最後の行でpをスクリーンにアウトプットするように命令しているが、このpが実際に表しているのは" personl computer" という文字列の先頭文字（この例ではブランク" "）の置かれているアドレスで、つまりメタファー的に表現すればpは文字列" personal computer" の居場所を指示（ポイント）していることになる。後は決められた約束に従って" personal computer" が文字を単位に次々とスクリーンにアウトプットされていく。

BASICの場合のような文字列変数と違いポインターは、結果として自分自身は文字変数として働きながら同時に文字列をも指し示す、という働きを持っていることになる。今一つの文字を固体、文字の集合である文字列を固体の属性（属性とはあるものの集合と規定できることに注意）、と置き換えれば、固体（文字（のアドレス））を指示しながら同時に属性（文字列）を指示しているポインターという考え方が固体の集合をベースに構築されているモデル理論にいかに反映されるべきか、筆者の考えがある程度察しがつくと思う。もちろんC言語のポインターとまったく同じとはいかないが、要はポインターのポイント機能を固体タイプ(e)のポインターに持たせるということになる。属性を固体タイプのポインターで指示したいわけである。今一度メタファーに立ち戻れば属性を固体の観点から理解・認識したいわけである。

7　今タイプ e の表現として

(16') i. *j*, m,. . .

ii. *x*, *y*,. . .

iii. **p*, **q*,. . .

を論理表現の中に認めることにしてみよう。(i) は固体定項、(ii) は固体変項、(iii) はポインターである。[6] またbe nice, be home, be pregnant, want, knowに対応する論理表現は次のようにタイプづけされているとする。

(17) i. *nice, home, pregnant*. . .　〈e, t〉（属性（固体の集合））

ii. *want,know*. . . . .　〈e, 〈e, t〉〉（固体から属性への関数）

この時次のような式は全て整合的であるといえる。(*want* (*x*, *y*) 等は*want* (y)(x) の省略表現)

(18)	式	cf.
	a. *nice* (*j*)	John is nice.
	b. *nice* (**p*)	Something is nice.
	c. *want* (*j*, *m*)	John wants Mary.

d. *want* (*j*, **p*)　　John wants something.

e. *know* (*j*, *m*)　　John knows Mary.

f. *know* (*j*, **p*)　　John knows something.

（C言語のシンタックスとはポインターの用い方が違うことに注意）ポインターはその内容が宣言されていなければ（18）のように英語の“something”等が意味的に対応すると考えられる。次にポインターが何かをポイントしている場合を“===〉”で表記し、⒅を基に本稿で取り上げたいくつかの文をある程度の論理表記として示していけば次のようになる 5（“---〉”は「として翻訳される」を示す）。

(19)

a. John is nice. ---〉 *nice*(*j*)

b. To be home is nice.

---〉 *nice* (**p*)

**p* ===〉 *home*（または λ*xhome* (*x*)）

c. John wants Mary. ---〉 *want* (*j*, *m*)

d. John wants to visit Mary.

---〉 *want* (*j*, **p*)

**p* ===〉 *visit Mary*（正確には λ*xvisit* (*x*, *m*)）

e. John knows Mary. ---〉 *know* (*j*, *m*)

f. John knows that Mary is pregnant.

---〉 *know* (*j*, **q*)

**q* ===〉 *that Mary is pregnant*

（正確には*pregnant* (*m*)）

ポインターを含む上記の論理表記は更に厳密には命題部分とポインター部分の順序対と考えることもできる。このことも含め（18),（19）の論理式の意味解釈を規定する作業が残っているが、“===〉”がポインター的にモデルの構成要素をもとにして規定できる集合等を差していること、及びその指し

方はLakoff and Johnson（1980）流のメタファー的認識の反映であると考えることができること、という二点が確認されれば本稿の目的は達成されたことになる。数多くあるであろう問題点の考察は次の機会に取り上げてみたいと考えている。

8　本稿のおおまかな内容を口頭発表[1]した時、金水敏氏からの質問でモデルを形式意味論における一般的なモデルとして考えているのか、それとももっとメンタリスティックなものと考えているのか、といった趣旨の質問を受けた。ことは非常に重大なので今後の研究方向との関係からもここでも一言付言して本稿を終えることにする。その時も同氏にもそう答えたが筆者は形式意味論におけるモデルはメンタルモデルでなくてはいけないと考えている。確かに我々は言語を用いて言語外の世界、例えば現実世界を認識しているのであるが、例えば標準的な形式意味論におけるモデルの要素に対応したものが我々の概念世界にもあって、実はその概念世界の中でさまざまな意味的現象が起こっているのではないかと考えている。今風に言えば情報はメンタルモデルを基盤にプロセスされている、とでも言える。もちろんそのモデルをどう考えるかは多種多様の可能性があるが、現時点では素朴に固体概念の集合みたいなものを考えている。メタファーが人間の概念形成や思考の中心的役割を果たすとするLakoff and Johnson（1980）の考え方に少なからず同調する本稿の立場からすれば、ポインターはまさにこの概念レベルでのポインティングであり、そのポインティングはメタファー的な理解として（8a）や（8b）のような認識パターンを可能にしていると言える。少なくとも現在の筆者はこのように考えたいと思っている。

注

1　この小論は1990年11月25日大阪大学における日米文科系学術交流研究集会・第27回待兼山ことばの会において開催されたメタファー研究集会で発表された内容の後半部分にあたる。その前半はおおむね杉本（1988）に相当する。その意味でこれは杉本（1988）の続編ともいえる。

杉 本 孝 司

2　メタファーとは全く無関係の意味公準も存在することに注意。それは例えば次の i から ii を論理的に含意するような場合。

i John walked slowly

ii John walked

同様の指摘は杉本（1988）注（９）参照.

3　もちろんメタファーは（8b）の場合であっても実際にはもっと複雑に関与していると考えられるがここでは簡略化して図示することにする。メタファーの経験的基盤（experiential bases of metaphors）に関してはLakoff and Johnson（1980），pp19ffなども参照されたい。

4　繰り返しになるがChierchia（1984）はまさにこの問題と取り組んでいることになる。

5　今、内包・外延の区別は取り上げないことにする。

6　ポインターの下位分類として固体ポインター、属性ポインター、属性の属性ポインター等を設ける必要があるかもしれないが、この点に関しては今後の課題としたい。このような下位分類を設けるにしても、シンタックスの上ではポインターはすべてタイプｅと規定されることに注意。ｍやｊが指示しているものはそれぞれ固体ポインターの値と考えてもよい。こう考えて行けばシンタクティックな項構造はすべてポインターからなると考えることもできる。筆者はこのようなシンタックスにもとずく形式意味論をポインター意味論（Pointer Semantics）と呼ぶことにしているが、この包括的な取扱はいずれとりあげたいと考えている。60－70年代式に文献表をあげるとすればこれは

Sugimoto, Takashi (To be written) “Pointer Semantics.”

ということになる。

参考文献

Carlson, Gregory N. (1977) *Reference to Kinds in English*, University of Massachusetts (Amherst) Ph. D. dissertation, distributed by IULC. (N. Y. : Garland)

Chierchia, Gennaro (1984) *Topics in the Syntax and Semantics of Infinitives and Gerunds*, Ph. D. dissertation, University of Massachusetts. (N. Y. : Garland)

Dowty, David R. et al. (1981) *Introduction to Montague Semantics*. Dordrecht, Holland: D. Reidel Publishing Company.

Lakoff, George (1987) *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*. Chicago: The Unviersity of Chicago Press.

Lakoff, George and Claudia Brugman (1986) "Argument forms in lexical semantics." *BLS 12*, pp. 442-454.

Lakoff, George & Mark Johnson (1980) *Metaphors We Live By*. Chicago: The University of Chicago Press.

白井賢一郎 (1991) 『自然言語の意味論』産業図書.

Sugimoto, Takashi (1988) " A Program for formal semantics and metaphors." 『英米研究16』, pp. 167-190. 大阪外国語大学・

Taylor, John R. (1989) *Linguistic Categorization: Prototypes in Linguistic Theory*. Oxford: Clarendon Press.

Turbo C User's Guide (1988). Borland International, Inc.