

## 日本語理解システムにおける文脈処理のモデル

横尾英俊 (山形大学工学部)

## §1. まえがき

自然言語処理の分野では単一文理解の水準がかなり進んだにもかかわらず、文の連続体である談話(会話や文章)の処理については多くの問題が残っている。これまでの実働システムの中では、長尾等<sup>1)</sup>が指示詞や省略語の解釈に成功しているが、単一文処理方式に単に文脈情報を取り込むだけではない、統語・意味文脈を有機的に結びつけた方法論はまだ模索の段階にある。

筆者は童話<sup>2)</sup>を対象にした物語理解システム制作の上で、文脈情報の特徴的な現われ方を分析し、それをアルゴリズムにまとめることを試みている。本論文は限定した日本語に対する文脈処理モデルの報告である。

## §2. 文脈理解の本質と処理の基本方針

文脈理解でまず問題となるのは、連体詞や代名詞の解釈および省略語の補足である。例(1)においては「結婚する」の主体を、(2)においては、「彼女」の指している具体的な対象を知らなくてはならない。これらは、「花子-結婚する」、「花子-彼女」というつながりの

- (1)<sup>注1)</sup> 花子<sup>注1)</sup>がもうすぐ会社をやめるらしい。  
結婚するのかな。  
(2) 花子がもうすぐ会社をやめるらしい。  
彼女、結婚するのかな。

の理解としてとらえることができる。文脈理解とはこのようなつながりの必要性を知って、連結操作を具体的に施すことに外ならず、(a) 連結操作が必要なことを知る (b) 連結操作を実行する、という二項目を文脈理解の本質とみることができる。人間がこの本質をどう扱っているかについての反省をもとに、ここでの文脈処理の方針として次の四項目をあげることとする。

(i) 意味的なかかり受け関係と構文解析

談話内の文に特徴的なことは、文脈の中にあるために、独立した単一の文にくらべて不完全な形態のもの(不完全形態文)が多いということである。ところが、これまでに開発されたATN<sup>3)</sup>や拡張LINGOL<sup>4)</sup>などの教理的構文解析手法は、統語構造に意味解釈を駆動させているために、不完全形態文まで含めようとする規則数もあいまい度も増加しシステムの記述がそれだけ困難になる。このような点から、統語規則の比重を小さくし、かわりに意味的なかかり受け関係を重視した構文解析を目ざすこととする。

(ii) 短期記憶フィールドモデルであること

意味的なかかり受けという観点では、文内のかかり受けも文脈を通しての連結操作も本質的な差はないことになる。問題は、どのような時期にこうした処理を施すかということになる。

人間はこれを次の二側面から解決していると思われる。ひとつは短期記憶的な

注1) 例文番号の右肩の数字は例文の出典となっている参考文献番号を示している。

フィールド上での操作であり、もうひとつは論理的な思考判断でもよぶべきものである。例えば(3)では「彼は」以後の第二文を読んだ後の論理的思考判断

(3) 太郎は次郎が病気であると思っていた。  
彼は……。

(4) 学校へ行く道にお寺があった。門をはいった所に大きないちょうの木があった。

を待たなければ「彼」を特定できないのに対し、(4)では「門を」を読み込んだ時点で「学校の門」ではなく「お寺の門」であることが理解される。つまり、文脈上での連結操作が短期記憶的なフィールド上で実行されている例なのである。

本論文では、論理的思考判断のモデルに先立ち短期記憶フィールド上でのモデルを設計する必要があると考え、これを中心にモデルを組み立てている。

### (iii) 辞書情報の利用

意味的なかかり受けを扱うには十分な辞書情報が必要であることは言うまでもない。ここでは人工知能における手続き付加の手法が使えるような柔軟な辞書構造を用いてこれを解決している。

### (iv) 言語構造の持つ情報の利用

文脈上での連結操作を行なう上で辞書情報以上に重要なのが言語表現の構造である。例(4)で「門を」を読み込んだ時点で連結操作が可能になるのは、(4)の表現にそれを可能にする構造があるからである。連接詞や代名詞もこのような構造をになう要素であるが、さらにこれほど明示的ではない場合についても連結操作が完全に行なわれるように分析を施している。

## §3 モデル

### 3.1 かかり受け関係の一般化

#### 3.1.1 辞書の構造

辞書情報の利用としてまず考えられるのは、各動詞の辞書にその格構造を明記する手法である。長尾らも文脈処理の基本としてこの手法をとり、空白の必須格が残っていればそれを文脈中から補うという形で文脈処理を実行している。ここでは、格情報をひとつのスロットで表現したユニット記法を辞書として用いることにする。例えば、「歩く」は格として有生(+animate)の動作主(AGENT)をとるので図3.1(a)のように表わされる。ユニット記法を動詞だけでなく、より一般的な語の記述形式として導入すると、(4)の「門」は図3.1(b)のように表わすことが可能になる。このように拡張すると、もはやスロットを「格」と呼ぶことは適切でないので、より広い意味でロール(role)と呼ぶことにする。

<歩く

(AGENT (+animate)) >

(a)

<門

(PART-OF (建築物)) >

(b)

図3.1 ユニット記法

例(5)及び(6)の第二文において、「誰が歩いているか」「何の温度か」を知るには、文脈中

(5) 太郎がいる。のろのろと歩いている。

(6) お湯がある。温度は40度である。

からロールの値として適切なものを探するという同一の手続きを行えばよい。このような動詞と格要素の関係をユニット記法上で拡張したものをかかり受け関係の一般化としてとらえ、これを基礎に文脈処理を行なうことにする。

#### 3.1.2 要求度

動詞の格をその動詞が述語として意味を完結させるために不可欠なもの(必須格)とそうでないもの(自由格)とに分ける立場では、空白のままの必須格要素を文脈中から探すということで連結操作を行なうことができる。しかし、ここのように格をロールとして拡張した場合には必須格の概念も拡張する必要が出てくる。さらに、(5)(6)に対して、連結操作を必要としない(7)(8)のような例 (7) 歩くことは健康に良い。を考えると、動詞の場合も必須か否か (8) 温度とは分子運動の程度である。というだけでは不十分で、むしろ(5)(6)と(7)(8)との言語表現の構造の違いこそが本質的であると考えられる。

つまり、各語がどのようなロールをどの程度必要とするかはそれぞれ決まっているが、文中ではその文の構造に応じて程度が変化すると考えるべきである。ロールをどの程度必要とするかの度合をロールの要求度と呼ぶことにすれば、語に与えられた各ロールの要求度が言語表現によって変換を受け、その結果としての要求度がある閾値を越えた時に連結操作が実行されるのである。

要求度の高いものとしては動詞の必須格をあげることができる。これは、(7)のような例を除けば、言語表現による変換を受けてもほとんどの場合文脈上の連結操作の対象になる。また、いわゆる関係名詞と呼ばれるものの時間や場所を示すロールも高い要求度を持っている。「前」「中」「上」などは「何の前」であるか等を知って初めて意味が完全になるのである。さらに、「音」「こおい」「気配」や(6)の「温度」なども「何の音」であるか等を強く要求するものである。図3.1(b)における「門」のPART-OFロールの要求度はこれらにくらべやや低いと考えられる。

要求度等に基づくつながりの良さの判定は、文内のかかり受け解析でも文脈上の連結操作でも、同一の単位を対象にしてなされる。どのような場合に、どの程度のつながりの良さを持つかは、ユニット記法でのスロットの第二項に柔軟に記述することにする。これは、パーカ MELING<sup>[5]</sup>での手法と全く同じである。

### 3.2 言語構造の持つ情報

ロールの要求度を数値として一般化するのは今のところ困難であるので、どのような場合に連結操作を駆動する閾値を越えるのかを考察することにする。

#### 3.2.1 言語と指示対象

人間が言葉を理解するということの第一点(9)<sup>[6]</sup>は、その指示対象を明確に意識することである。(9)においては、「サドル」が「きのう買った自転車のサドル」を指しているということを意識して初めて理解が成立する。そして、このような理解を誘導している要因は「サドル」という語の現われ方にある。(9)に類似した例として(6)をあげることができるが、これらに共通していることは、「サドル」「温度」が共に「は」を伴って文頭にあること、前文にそれらを部分あるいは属性として持つ「自転車」「お湯」がそれぞれ存在することである。「は」の用法には複雑な側面があり簡単に論ずることはできないが、(6)や(9)のように、文頭に「は」を伴った名詞句があり、しかも前文にその広義の上位概念<sup>[注1]</sup>が存在する場合には文脈指示<sup>[注2]</sup>の用法である場合が多い。つまり、このような名詞句に対しては連結操作を施す必要があるのである([手続きF]参照)。

ところで今の場合、連結の相手は「きのう買った自転車」であり、これは既に

注1] 集合論的包含、地理的包含、部分全体、所有関係、属性などを広義の上下関係と仮称している。

文脈上に登録されていなければならない。つまり、第一文の「自転車」は文脈上に特定の自転車を導入していると考えるべきものである。言語のこのような用法をここでは alpha 類と呼び、特定の指示概念を持たない場合は beta 類と呼ぶことにする。この分類によれば、(6)での「温度」による指示対象は alpha 類であり、(8)の「温度」は beta 類に属することになる。文頭において beta 類の指示概念を表わす語に「は」が附加された場合は総称<sup>判</sup>の用法であることがみい。以後は、言語表現 e の指示対象の類を type(e) で表わすことにする。

一般に固有名詞や話し手と聞き手の間で指示対象が一意に特定化されているものは、辞書の中で既に alpha とマークし、それ以外のものは beta とマークする。beta とマークされた語が文内での用法に従い、alpha 類となるか beta 類のままであるかがここでの文脈処理の重要な鍵である。

### 3.2.2 文脈情報の記憶構造

文脈との連結操作が必要であることがわかれば、連結の相手が文脈中に検索される。この時、文脈は連結の可能性の大きいものほど取り出しやすい構造になっていることが望ましい。可能性の大きいものには、その時点で焦点となっているもの、語順が後方であるものなどがある。ここでは、文の上に提示された概念のある規則に従ってスタックすることで文脈を構成する〔手続き E〕。

### 3.2.3 連結操作

以上の考察や言語学的な知見をもとに、データの分析結果をできるだけ簡単な手続きの形にまとめる。まず体言句に対し図 3.2 の句構造を仮定し、その上での手続きを述べる。

#### 〔手続き A〕タイプ決定手続き

1. M = 「は」なら、「は」解釈手続きへ。
2. N が代名詞なら、連結操作を実行し 12へ。
3. type(N) = alpha なら、type(NOM) ← alpha とし 12へ。
4. DET が空なら 7へ。DET が「その(あの・この)」であれば 9へ。さもなければ 5へ。
5. DET に alpha 類のものを含んでいるなら、type(NOM) ← alpha とし 12へ。DET が「NOM'+の」でかつ N が関係名詞類なら 6へ。さもなければ 7へ。
6. NOM' に連結操作を施し、type(NOM) ← type(NOM') とし 12へ。
7. M = 「が」なら 8へ。さもなければ 9へ。
8. NP が従属文中にあれば連結操作を施し 12へ。さもなければ 11へ。
9. M = 「の」なら type(NOM) ← type(N) とし 12へ。さもなければ 10へ。
10. N に連結操作を施し 12へ。連結が失敗したならば 11へ。
11. type(NOM) ← alpha とし NOM に SMT とマークし 12へ。
12. スタック手続きへ。

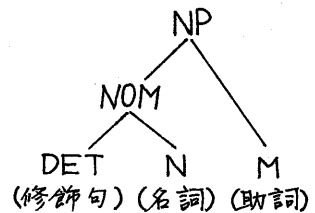


図 3.2 名詞句の構造

先にふれたように、「は」の用法には複雑な側面があり、「は」解釈手続きを一般的にまとめるのは困難である。本論文では、次節で述べる制限された日本語に対する「は」解釈手続きを §5 で論ずる。スタック手続きについても同様である。

#### 〔手続き B〕連結操作 (対象 X)

代名詞ならばその素性に合うもの、普通名詞なら、それと同一か或いは広義の

注1 文脈指示、総称：久野<sup>判</sup>などの用語である。

上位概念を表わす語によって指示された概念(R)を文脈中に検索して連結し、 $type(X) \leftarrow type(R)$ とする。文脈中に適当なものが見つからない場合、連結操作は失敗したと言う。 □

次に用言句に対する連結手続きを考える。動詞の必須格は要求度の高いロールであるが、自由格でも(10)のような例では文脈中から補うことが必要になる。つまり (10)<sup>21</sup> 小さな島がありました。まっか  
り「咲いている」に対し「場所」は必須 なつばきが咲いていました。  
のロールでないにもかかわらず、第一文  
の「小さな島」を補わなくてはならない。このような事情を一般的に説明するには、ロールの要求度を  $D(\cdot)$ 、連結操作を駆動する閾値を  $\theta$  とした時

$D(\text{必須格}) > \theta > D(\text{自由格})$  : 一般的関係

$D(\text{自由格}) + f(\text{文脈の構造}) > \theta$ : 自由格に連結操作が必要となる場合となるような  $f$  について考察しなければならない。ここではその最も単純な形式として次の格補充手続きを提案する。

#### 【手続きC】格補充手続き

必須格が空白であれば文脈中から補う。空白の自由格があり、文脈スタックの一番上の要素が SMT とマークされていて且つその自由格を埋めるのに適当であれば補充を行なう。 □

もちろん、この手続きだけでは(5)(7)のような例の區別を行なうことはできない。これらを含むした手続きは今後の課題である。

なお、用言句についても体言句と同様、タイプ決定手続きを行ない文脈スタックへのスタックを行なうが、詳細は省略する。

## §4 シミュレーションのための日本語の簡略化

以上の考察を計算機でシミュレートする場合、文脈処理に先立って考察しなければならないことが数多くある。「わかり書き法」ひとつにしてもまだ完全な解はなく、ここでの理論のみを試すにはある程度わかり書きされた入力が必要となる。このようなことから、入力日本語に制限を加えることにする。

### 4.1 制限された日本語の本質的構造

まず、主な制限を列挙する。

- R1) 常体(ダ体)であること。
- R2) 訓令式ローマ字表現であること。
- R3) 体言句はわかり書きされた一個の助詞を最後に持つこと。
- R4) 助詞「は」を最後に持つ名詞句は文頭にあること。
- R5) 体言句と用言句はわかり書きされていること。
- R6) 表現の水準が同一であること。つまり、会話や引用を含まないこと。

これらの制限によって、例(10)は

TIISANA SIMA GA ATTA. MAKKANA TUBAKI GA SAITEITA.

と入力されることになる。以上のほかにもいくつかの制限を設けているが必ずしも本質的なものではない。

さて、このように制限された日本語の構造について考えてみると、統語的にかかり受け関係は本質的に次の四種であることがわかる。

M1) 体言→体言 TARO NO TITI

M2) 体言→用言 TARO GA IRU

M3) 用言→用言 MAKKANI SAKU

M4) 用言→体言 TIISANA SIMA

これらの特徴づけているものは、M1, M2では助詞、M3, M4では用言の活用形である。このような例に現われる助詞、活用語尾を、特徴を明示している印という意味でマーカと呼ぶことにすれば、マーカ以外の自立的な要素(F要素と呼ぶ)はマーカによって構造的に結びつけられていることになる。言わゆる連伴詞や副詞はそれ自身がマーカとしての情報を含んでいる語と言える。

M1	no			(の)
M2	ni			(に)
	ga			(が)
	kara			(から)
	made			(まで)
M3	de			(で)
	i,	ku,	ni	(ク)
	ite,	kute,	de	(テ)
	reba,	kereba,	nara	(バ)
	itara,	kattara,	dattara	(タラ)
M4	ruto	ito,	dato	(ト)
	ru,	i,	na	(ル)
	ita,	katta,	datta	(タ)

表4.1 マーカの例

マーカとしてどのような要素を認定するかは、立脚する文法理論やシステムの設計思想に依存すると思われるが、ここでは入力文をF要素とマーカとの対立形式でとらえ易いように表4.1のように定めた。表4.1において、<sup>b</sup>は直前が母音ならa、子音ならbが省略されることを意味する。この結果、「夏になると」は、マーカを小文字で表現した場合、「NATU ni NAR ruto」と表わされる。

日本語の文をF要素とマーカの連続体としてとらえると、マーカによって決定される統語的<sup>a</sup>なかり受け関係の可能性の中から、意味および文脈の上で最適なものを選び出すことが主たる問題となる。

#### 4.2 助動詞等の扱い

言わゆる補助用言や助動詞は、それが附加された主動詞と共にひとつの自立的な要素を構成していると考えられる。その結果、このひとまとまりの要素が、固有のマーカで固有のルールを取ることになる。例えば「置く」は、「本を置く」のようにマーカ「を」を伴って“対象”をルールとして取るが、「置いてある」は「が」を伴って“対象”ルールを取る。このような解釈を可能にするために、補助用言や助動詞は主動詞とわから書きさしないうままに入力することにする。

### §5. システム設計

図5.1は本モデルによる処理の流れの概略を、拡張LINGOLやMELING上のシステムとの対比によって示したものである。

#### 5.1 部分構文樹(PPT)の生成

入力文は図5.2に示す手順によってF要素とマーカの対(「統語対」とよぶ)の列と見なされる。統語対間の依存関係がマーカの種類に応じて調べら

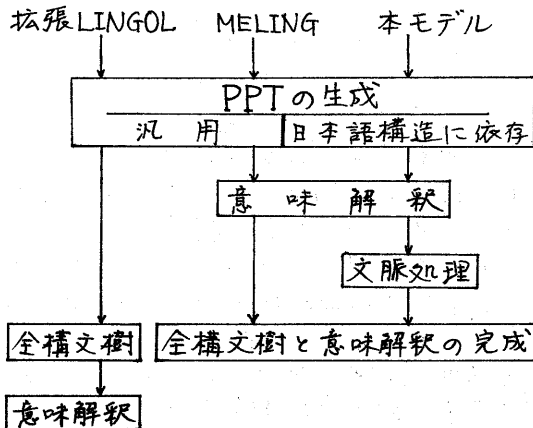


図5.1 処理流れの比較

れ、可能な部分構文樹(PPT)全てに対し意味解釈が実行される。さらに、各PPTの最右のF要素をPPTの主要語とよぶことにすると、主要語が体言の場合は【手続きA】により、用言の場合は【手続きC】により文脈との連結操作が施され、各PPTに応じた文脈スタックが形成される。各PPTは意味解釈の副作用として、それぞれの「もっともらしさ」を算出し、あらかじめ指定された個数のPPTだけを残し、「もっともらしさ」の低いPPTはそこで棄却される(15)参照)。

【手続きD】は、制約R7が課せられた場合のPPTの具体的な生成法を示している。

- R7) 統語対間の依存関係(かかり受け)が非交叉であること。
- 【手続きD】  
 recursive procedure GEN-PPT(PPT, PAIR, i)  
 1. if NULL(PPT), return PPT; + CONS(PAIR, PPT)  
 2. if INSTP(CAR(PAIR), CAR(PPT)), return  
   PPT; + CONS(PAIR, PPT):  
   PAIR + FILL-ROLE(PAIR, CAR(PPT)):  
   GEN-PPT(CDR(PPT), PAIR, i+1)  
 3. return PPT; + CONS(PAIR, PPT)

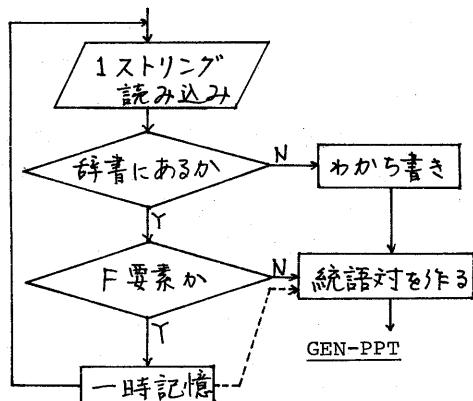


図5.2 統語対の読み込み

PPTはスタック構造として表現され、一番上に積まれた要素をCAR(PPT)、それを取り出した残りをCDR(PPT)で示してある。PPTの各スタック要素は、統語対と同じように対構造(PAIR)をなしている。GEN-PPTは、今読み込んだ統語対をPAIRとし、それまでにできている何個かのPPTに順に適用される。FILL-ROLE(P1, P2)とは、P1のF要素のルールにP2から値を代入するもので、FILL-ROLE(「食べた」(りんごを))であれば、(食べる<sub>1</sub> た)を返す。但し「食べる<sub>1</sub>」は

<食べる<sub>1</sub> alpha (OBJECT = りんご)>

という内部表現で表わされる意味ユニットと呼ばれるものである。このようなロールの値代入の可能性をF要素とマーカの種類から決定する述語がINSTPである。ロールに値を代入する際には、手続き付加の手法により「点数」が計算され、それをもとに各PPTの「もっともらしさ」が算出される。

5.2 文脈スタックの形成

各PPTはそれぞれの解釈に応じ文脈情報をスタックとして表現している。PPT生成はひとつの句構造の仮定に対応し

【手続きA】又は【手続きB】が実行され、その中で文脈スタックへのスタック手続きが呼び出される。【手続きE】はPPTの最上のスタック要素を図5.3のように考えた時の文脈スタック(CONTEXT)へのスタック手続きである。

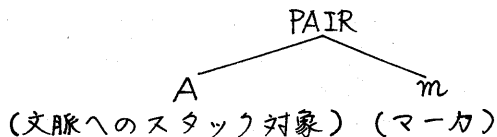


図5.3

【手続きE】 スタック手続き

1. AがTHとマークされているならそのままスタックレクへ。

2. CAR(CONTEXT)が、THとマークされているならば、CDR(CONTEXT)にまずAをスタックし、次に最上部にCAR(CONTEXT)をスタックし直して7へ。
3. AがSMTとマークされているならばそのままスタックして7へ。
4. CAR(CONTEXT)がSMTとマークされているならば、CDR(CONTEXT)にまずAをスタックし、最上部にCAR(CONTEXT)をスタックし直して6へ。
5. Aをそのままスタックして6へ。
6.  $m \in M3UM4$ であれば、A及びそのルール要素より下にスタックされているもののうちSMTとマークされたもの(がもしあれば)のうち一番上のものを取り出し、CONTEXTの最上部にスタックし直す。
7. END □

1や2に現われるTHとは、次の「は-解釈手続き」によってマークされるものである。「は-解釈手続き」は【手続きA】の中で呼び出される。

#### 【手続きF】 は-解釈手続き

1. Nが代名詞なら連結操作を実行しNOMにTHとマークして10へ。
2.  $type(N) = \alpha$ なら  $type(NOM) \leftarrow \alpha$  としNOMにTHとマークして10へ。
3. NOMと同一主名詞(N)による指示対象が文脈スタック中にあるならば5へ。
4. Nの上位概念、上位部分を表わす語による指示対象が文脈スタックにあるならば7へ。なければ8へ。
5. 述語の時制形がタ型ならNOMと文脈との間に連結操作を施して10へ。
6.  $type(NOM) \leftarrow type(N)$ とし、NOMにTHとマークして10へ。
7. NOMに連結操作を施し、NOMにTHとマークして10へ。
8. Nの意味ユニットFnでまだ値の埋まっていないルールと同一のルール名を持つルールRnが前文の意味ユニット内にあり、しかもそれが値を有しているならば、その値をFnのルールに代入して10へ。
9. Rnが値を持つなければ、それを先行詞とする関係節に対応する値をつくりその値をFnのルールに代入して10へ。
10. スタック手続きへ。 □

#### 5.3 解析例

簡単な例として(11)に以上の諸手続きを施してみる。(11)では「両手」が「花子の両手」であること、「喜んだ」対象が「雪が降ったこと」もしくは「雪」であることを理解しなければならぬ。図5.4にそのことを表現した意味ユニットが、図5.5にはそれが得られるまでの処理過程の一部が示してある。図5.4の意味ユニットでは、時制情報の表現などを全て省略している。

図5.5では、実線の枠でPPTを、破線の枠で文脈スタックを表現している。まず統語対「(冬休みに)」が読み込まれるとPPT<sub>1</sub>が作られ、【手続きA】が実行される。【手続きA】の11によって「冬休み」がSMTとマークされ、続いて【手続きE】によって文脈スタックが生成される。PPTはひとつの文の処理を終える毎にすべてフリーされ、「(喜んだ)」を読み終えるまで処理が続けられる。処理が終了した時点で「もっともらしさ」の最も高いものが解釈結果となる。例ではPPT<sub>10</sub>に図5.4に示した意味構造が表現されている。ここで注意すべきことは、「花子」「両手」が同一の文にありながら文脈処理によって「花子の両手」と推



<喜ぶ<sub>2</sub> alpha

(AGENT = 花子)

(OBJECT = 降る<sub>2</sub>)

(MANNER = あげる<sub>1</sub>) >

<あげる<sub>1</sub> alpha

(AGENT = 花子)

(OBJECT = 両手<sub>1</sub>) >

<降る<sub>2</sub> alpha

(OBJECT = 雪)

(TIME = 冬休み) >

<両手<sub>1</sub> alpha

(POSSESSOR = 花子) >

図5.4 (11)の意味表現

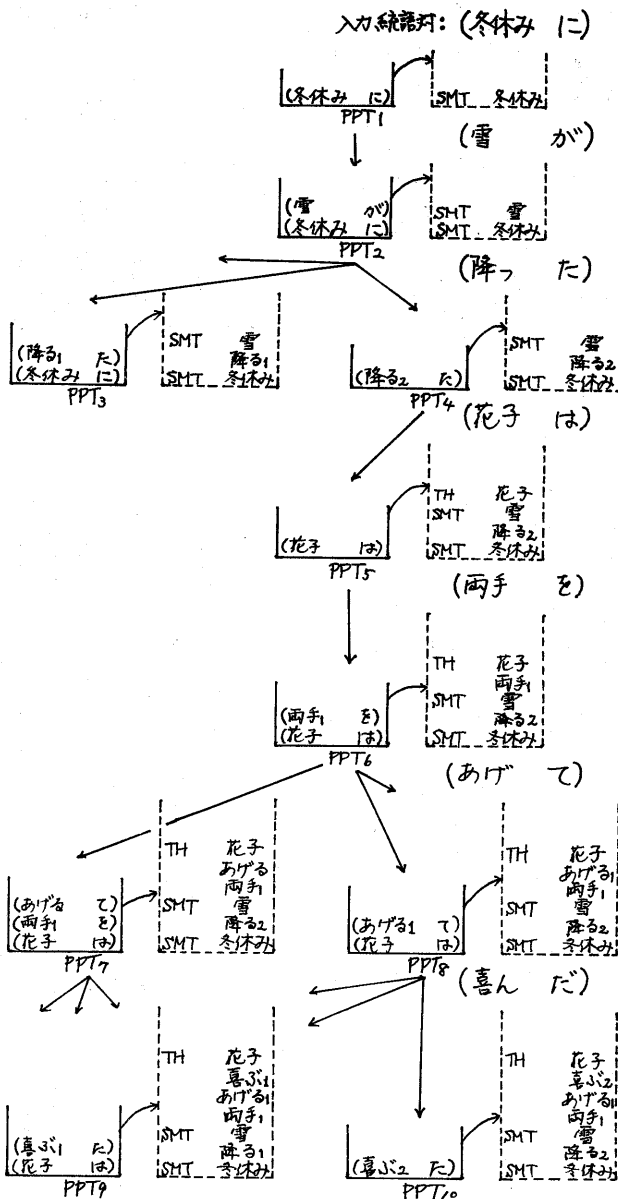


図5.5 (11)の解釈過程

論されている点である。つまり、句構造的な統語規則よりも、語順や語の導入のしかたなどの文脈情報によって文の解釈が方向付けられていると言える。

### 5.4 各手続きの意味付けについて

これまで、各手続きを、ほとんど合理的な説明を加えないままに提示してきた。各手続きとも、データの分析から帰納的に「工夫」されたものであり、個々の観察事項を例をあげて詳細に述べるのは困難だからである。このような手続き構成の性格上、各手続きはまだ粗い近似であり、より精密なものにする必要があることは言うまでもない。問題は、この方向での精密化が本質的解決につながるかどうかということである。

手続きの理論的裏付けを試みると、言語理論において文脈構造の特徴的な現われとみなされている「は」と「が」や「新情報」と「旧情報」などの使いわけが、手続きの中に反映されていることがわかる。例えば【手続きA】のSMTは、新情報の概念をマークしているものであり、従って【手続きE】では文脈の最上部にスタックされるように意図されている。もちろん、SMTというマークが完全に望ましくなされるわけではないので、満足すべき解釈結果が得られない場合もある。

例文(12)は独立した単一の文 (12) 月が雲に隠れて見えなくなった。  
として成立し得る文であり、 (13) 花子の影が月の光にうつし出されていた。  
「雲に隠れた」対象および「

見えなくなった」対象として「月」を認定することができる。しかし、(12)の前に(13)のような文で作りに出された文脈があると、「見えなくなった」対象は「月」ではなく「花子の影」と解釈する方が自然になる。もし「花子の影」にSMTとマークすることができれば、以上述べてきた右手続きによってこれらの解釈は完全に可能である。しかし、[手続きA]では「花子の影」にSMTとマークすることはできない。「の」や「影」の機能的な作用の分析がまだ不十分だからである。それでもこのような例は各手続きの精密化の可能性とその意義を十分示唆していると言えよう。

## §6 まとめ

本論文では制限された日本語に対する文脈処理のモデルを中心に報告した。従来の考えの多くは、単一文処理モデルと談話処理モデルとの間に大きな不連続性があり、人間の処理様式と比較した場合かなりの不自然さがあった。特に統語規則駆動型のシステムでは、(12)のような文を文脈によって解釈し分けようとした場合、(12)の統語構造自体に二通りのものを仮定しなければならず、統語構造生成を文脈によって制御するのが困難であった。

これに対し本モデルでは、単一文処理の基本に

(a) スタックを用いた係り受け解析

という単純な方式を用いながらも、逐次

(b) 文脈の生成と参照

を行なうことによって柔軟で自然な処理ができています。今後は(a)(b)両面からモデルを精密にしなければならぬが、(a)に関しては池田<sup>[1]</sup>に、より精巧なモデルがあり、(b)についても[1]や[6]を参考に本モデルの立場から結合をはかり、ここでは扱わなかった後方照応等も処理できるモデルに仕上げて行く必要がある。

謝辞 日頃御指導御助言いただく片桐理和教授に感謝する。なお、本論文は筆者が東京大学大学院に在学中の、伊理正夫教授、白井英俊氏の指導による修士論文研究に基づくものであることを付記し、両氏に感謝する。

## 参考文献

- [1] 長尾, 辻井, 田中 「意味および文脈を考慮した処理—文脈を考慮した処理」 情報処理, Vol.17 No.1 (1976), pp.19-28. [2] 浜田広介 「浜田広介會話集」 新潮社 [3] Woods, W. A. "Transition network grammars for natural language analysis," CACM, Vol.13 (1970), No.10, pp.591-606. [4] 電総研推論機構研究室 「拡張LINGOL」, 1978. [5] 白井, 横尾 「統語処理と意味解釈を同時に行うシステムについて」 情報処理学会計算言語学研究会 21-2 (1980). [6] 田中, 「談話の理解とメモリ・モデル(その1~3)」, 電総研集報, 第40巻(1976), 7, 9, 10. [7] 久野, 「日本文法研究」, 大修館, 1973. [8] Ikeda, T. "J-Analyser: Analyser of Japanese Sentences Based on Binding Structure Representation of Words," Journal of Information Processing, Vol.3, No.4 (1980). [9] Hinds, J. "Aspects of Japanese Discourse Structure," Kaitakusha, 1976.