

論文

意味および文脈情報を用いた日本語文の 解析—文脈を考慮した処理*

長尾 真** 辻井 潤一** 田中 一敏**

Abstract

In the previous paper⁵⁾, we explained the semantic descriptions of verbs and nouns. In this paper, we explicate how contextual information is organized in our system and how contextual information and semantic descriptions are combined in order to perform contextual analysis, for example, to fill up omitted words from preceding sentences.

In most of the artificial intelligence approaches to computer understanding of natural language, logical expressions are used to represent context, and rather complex logical operations are required to perform even a simple contextual analysis¹⁾. On the contrary we memorize context in the form similar to the short-term memory of human beings. According to several heuristic's the words in the memory are re-ordered in order to emphasize the theme words which express the key subjects of the sentences.

We describe the detailed construction of this memory. Several examples of complicated sentences parsed by our system are also shown in this paper.

1. まえがき

前論文⁵⁾では、単文内での名詞—動詞の格関係の処理及び名詞句の処理等について報告した。これらの処理は比較的局所的な単語間の係り受けに関する処理であり、統語情報に導かれた意味処理が中心となっていた。従って、省略された語や句を文***の境界を越えて補足するといった処理は述べなかった。

日本語においては、話題の中心になっている単語ほど省略されたり、代名詞化されることが多い。この省略語・句を前後の文から埋めることは、文章分析の質を高め、計算機による自然言語の理解には不可欠である。しかしながら、このような処理では、手掛りとなる統語情報が少なく、むしろ前の文章から生じた文脈

と、処理中の文の意味との関係が重要な役割を果たす。

本論文では文脈の表現形式及び文脈の参照のメカニズムについて述べ、これを用いた文脈を考慮した処理について報告する。最後に、これらの複雑な処理をとる文の分析結果について述べる。

2. 文脈処理の方針

人間の文章理解のプロセスを考えると、一応主要な解析の方向（理解の道筋）は、文頭から文末へと進んでいくと考えられる。しかし、現在理解しようとしている文において情報が欠けている場合には、以前読んだ文にもどるか、あるいは頭の中にできている文脈を参照することにより、欠落した情報を見出そうとする。この前文の参照によって満足のいく結果が得られなかった場合には、一応未解決な問題として保留し次の文の解析へ進む。そして、次の文中でこの保留された問題を解決する可能性のある単語なり句が出てきた時点で、実際に解決できるかどうかを調べ、解決できればこの問題を頭の中から消去する。しかし、未解決な問題も、いつまでも頭の中に残っているのではな

* Analysis of Japanese Sentences, by Using Semantic and Contextual Information—Context Analysis, by Makoto NAGAO, Jun-ichi TSUJII and Kazutoshi TANAKA (Faculty of Engineering, Kyoto University)

** 京都大学工学部電気工学第2学科

*** 本論文中で用いる「文」及び「単文」は、「動詞」等の用言によって統合されて一つの意味のまとまりを持つ単位をいう。従って、「。」で区切られた範囲内でも連体修飾句としての「文」や連用中止法で連続する「文」など複数個の「文」が存在することになる。

く、時間がたつにつれて薄れてしまう比較的浅いレベルの記憶であると思われる。

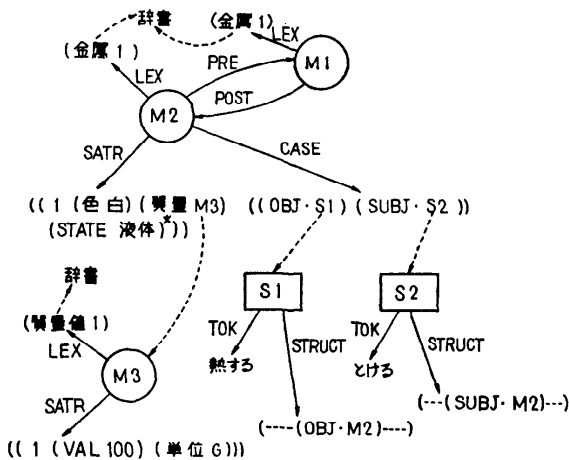
この様な過程を計算機上に実現するために我々がとった方針を以下に列挙する。

- (1) 文脈の記憶は名詞を中心とした形にする。
- (2) 1つの単文、名詞句等のある意味的まとまりを持った単位の処理が終了した時点で、文脈を考慮した処理を行う。
- (3) 文脈の表現形式は論理的操作に適した形式よりも、むしろ人間の短期記憶 (short term memory) を模した形式にする。
- (4) この短期記憶の形式は話題の中心になっている様な語ほど、簡単に取り出せるような表現にする。
- (5) 後文の参照が必要となった時は、未解決な問題として設定するだけで実際に後文の参照は行わない。後文が解析されてゆく時点で、その未解決な問題が解消されてゆくような機構にする。

3. 文脈情報の記憶

文の解析は、中心となる動詞を手掛りとして行ったが、その結果としての文脈はその文中に現われた名詞(対象物)についてどのようなことが言われたか、またそれらがどのように変化したかに注目して記述したほうが都合が良い。そこで、文中の名詞に LISP アトム M1, M2, ... (以後名詞アトムと呼ぶ) を割当て、このア

入力文: 「---金属がある。この白い金属100gも熱したらとける。」



(*) M2のSATRの(STATE 液体)はS2の処理後付加される。

Fig. 1 Usage of Noun-Atoms

Table 1 Information attached to a Noun-Atom

記号	意味
LEX	辞書情報へのリンク
SATR	文脈情報
CASE	文構造へのリンク
PRE	同一指示対象で以前に出てきたものへのリンク
POST	同一指示対象で以後に現われたものへのリンク
SMOD	埋込み文の動詞により形容された名詞の動詞へのリンク
PARA	並列する語とのリンク

トムの性質リスト (property list) から対応する名詞の全ての情報が得られるようにした。用いたリンクとその意味を Table 1 に、例を Fig. 1 に示す。

我々はこの名詞アトムに対して、次の2種類のスタックを作り、文脈を表現している。

3.1 M-スタック (名詞スタック)

一単文の解析が始まる前に、その文中の各々の名詞に対応する名詞アトムをリストにして M-スタックに積み、処理が終わった時点でこれに変更を加える。即ち M-スタックは現在までに実際に文中で言及された名詞のスタックであり、Fig. 2 の構造をもつ。この単文ごとに区切られたリストの中での名詞アトムは次の順序に従って並べられている。

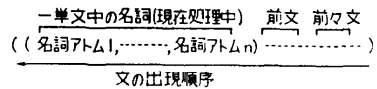


Fig. 2 M-stack

(1) 補足された名詞に対応する名詞アトム。前述のように、日本語では話題の中心となる名詞は一度文中に現われると、それ以降は省略されることが多く、特に連用中止法ではこれが著しい。処理プログラムはこの M-スタックを探索することにより省略語句を補うのであるが、その際補足された名詞は、その単文に対応するスタック中のリストの先頭に追加され、次の探索操作の際に優先度が高くなるようにされる。

(2) 動詞の格要素になった名詞アトム。これらの名詞アトムはさらに '人間以外の行為者格' '>' 直接目的格 '>' 間接目的格 '>' それ以外の格' の順に並べられる。

(3) 直接動詞の格要素にはならなかった名詞。

(4) (1), (2), (3) で同一の優先度にある名詞は、出現の逆順に並べられる。

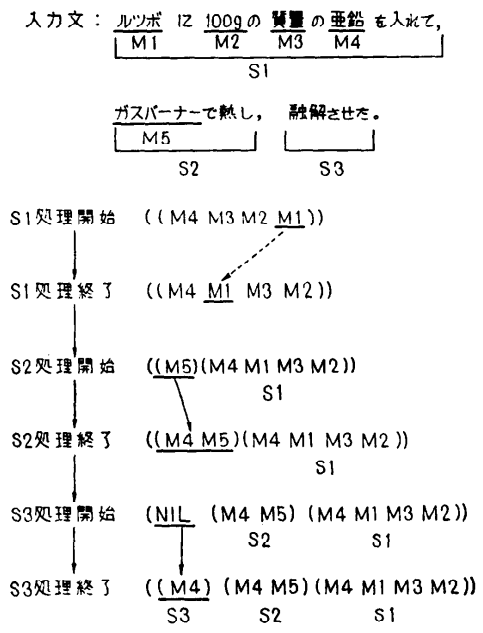


Fig. 3 Changes of M-stack

一例を Fig. 3 に示す。これによると亜鉛 M4 はすべての文に出現していることになる。

3.2 KM-スタック (仮説名詞スタック)

「水素と酸素を 2:1 の割合で混合し，この混合気体に……。」

「食塩 5g を 100 cc の水に溶かす。この水溶液は……。」

この様な文では、「この」という既出の語句の存在を要求する連体詞があるにもかかわらず，前文にはこれに対応する名詞が存在していない。前論文で述べたように動詞の意味辞書には，各動詞の格構造が記述されているが，上記のような例文を解析するには，これだけでは不十分である。即ち，各動詞の格構造は「現実世界に起り得る事柄」を表現していると考えられるが，人間はある「事柄」が起ると，その「事柄」に関与した「物」がどのように変化し，またその結果として何が生じるかを知っていると考えられる。我々の解析システムではこの種の知識を動詞の格構造に付加すること(3.3 参照)によって，「混合する」，「溶かす」という事柄が起ると，それぞれ「混合物」，「溶液」が生じることを知る。このあらたに生じた対象物もそれぞれ固有の LISP-アトムに対応づけられ，スタックされる。これらのあらたに生じた対象物は，必ずしも存在が保証されていないので KM-スタック (仮説名詞スタック) 上にスタックされる。KM-スタックも各

単文ごとのリスト構造になっており，一単文の処理が終了すると，その単文から発生が予想される名詞のリストがスタックされる。

3.3 文脈処理のための動詞辞書

1つの単文内での係り受け関係が決定されると，その単文を支配している動詞の表層格構造の各格要素が確定する。前論文⁵⁾で述べた評価関数により最も高い評価値を得た表層格構造が辞書内の記述に従って深層格構造に変換され，それと同時に，その事柄の side-effect が文脈に反映される。side-effect としては，

- (1) どのような事柄が次に生起するか
- (2) その事柄に関与している「物」がどのような変化を受けるか

が動詞辞書に記述されている。これを利用することにより文中には明示されていない事柄・状況を把握することができる。辞書内には次の2項目の記述がある。

- (1) CON: ある事柄が起こった時に帰結として起こる事柄
- (2) NTRANS: 事柄に関係した「物」に生じる変化

(1)の例として，「入れる」の辞書を Fig. 4 (a) に示す。「水を試験管に入れる。」が解析されると，表層格構造の目的格，場所格がそれぞれ「水 (M1)」，「試験管 (M2)」になる (M1, M2 は名詞アトム)。この場合，深層格構造がないので，表層格構造=深層格構造と解釈される。次に CON が参照され，(ある (SUBJ M1) (IN M2)) (「その水がその試験管の中にある」) が導き出される。

(2)では，新しい「物」を作り出す CREAT, 「物」の持つ属性の変化を示す ADD, DELETE が用いられる。

この例として，「とかす」の格構造を Fig. 4-(b) に示す。例えば，「食塩を水にとかす」では，第1の格構造で処理された結果が最高の評価値を得る。そしてその NTRANS を参照することにより，「溶質」がその食塩 (M1) で，「溶媒」がその水 (M2) である溶液が生じたことがわかり，それを KM-スタックにスタックすることになる。また，「イオウをとかす」では第2の格構造が採用され，その結果目的格に入ったその

- (a) 入れる→CF-1 (((ACT 人間) (OBJ 物質) (IN 容器)) (CON (ある (SUBJ OBJ) (IN IN))))
- (b) とかす→CF-1 (((ACT 人間) (OBJ 物質) (IN 液体)) (NTRANS (CREAT (溶液 (溶質 (* OBJ)) (溶媒 (* IN)))))) CF-2 (((ACT 人間) (OBJ 固体) (INST 熱する道具)) (NTRANS(ADD OBJ (STATE 液体))))

Fig. 4 Verb descriptions for contextual analysis

イオウ (M1) の属性 'STATE' に「液体」が入れられる。

4. 指示詞の解釈 (アナホーラの処理)*

4.1 連体詞による指示

指示連体詞による指示は、「その銅」の様に連体詞に続いて具体的な指示対象がある限定指示と、「その体積」が「その亜鉛の体積」を意味するような代行指示の2つがある²⁾。中学校化学の教科書で用例を調査した結果を Table 2 に示す。この用例を基にして作成した処理手順を Fig. 5, 処理例を Fig. 6(次頁参照) に示す。なお代行指示の用例はいずれも連体詞に続く名詞が relational word の性質を持つもので、論文³⁾ で述べた名詞のカテゴリ分類及びその意味記述を使うことにより処理される。

4.2 指示代名詞によるアナホーラ

指示代名詞が名詞句の中の一要素になっている場合は、連体詞の代行指示の場合と同じ処理がされるが、代名詞が動詞の格構造に直接入っている場合は、その動詞の格構造の参照が必要になる。

〔例〕 水素と酸素を混合し、それに火をつける。

この例文の場合、使用できる情報は「それ」についての格助詞「に」と、動詞「つける」の格構造である。「に」の表現し得る格の中で、「つける」の格構造内にある格は間接目的格であり、かつ「つける」の間接目的格の要求する名詞は「物質」である。従って、解析プログラムは、「物質」を上位概念としてもつ語

Table 2 Usages of 'KONO' and 'SONO'

用 例	頻 度
被修飾語と同じものが M-STACK にある	60
被修飾語の低位概念が M-STACK にある	14
被修飾語の上位概念が M-STACK にある	6
被修飾語と同じものが KM-STACK にある	3
被修飾語の上位概念が KM-STACK にある	9
被修飾語が属性名詞で、それを属性としてもつ名詞が M-STACK 中にある	10
被修飾語が機能名詞で、M-MFUNC のチェックが成立する名詞が M-STACK にある	14
被修飾語がサ変名詞	3

* 連体詞及び代名詞の指示の仕方には、「現場指示」と「文脈指示」がある。T. Winograd は前者に近い観点から処理を行っているが、我々は主として後者に焦点をあてている。

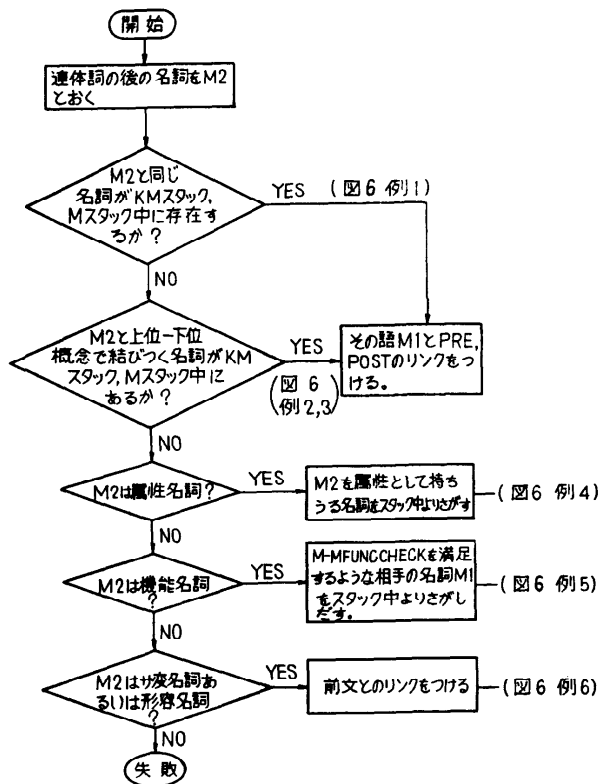


Fig. 5 Algorithm of processing 'SONO' and 'KONO'

を M-スタック, KM-スタックから探索する。M-スタック, KM-スタックの探索順序は、「ある事柄に関与した物の集合が M-スタック, その結果生じた物が KM-スタックにある」ため、現在処理中の文により近いのは KM-スタック中の名詞であると考えられる。従って、(前1文の KM-スタック) - (前1文の M-スタック) - (前2文の KM-スタック) - ……、の順で前3文まで探索される。この例では、KM-スタックより「混合物」が条件に適合するものとして引出され、結局「それ」=「混合物」と判断される。

5. 省略語の推定

省略語を処理する際に必要な操作は、(1)省略が行われたことを知る。(2)省略された語がうける意味的制限を知り、その制限に合致する語・句を前後の文から探索すること、である。単文あるいは名詞句のような統語上の単位は、一般に意味的なまとまりの単位になっている。従って、単文や名詞句の解析が終了時点で、それらが意味的なまとまりを形成するために必

- 例1 銅が100gある。この銅とイオウを試験管に入れる。
同一の単語
- 例2 ある金属 10cm³の質量は75gであった。この物質の密度を求めよ。
上位-下位概念
- 例3 食塩を水にとかす。この食塩水に硝酸銀水溶液を加える。
溶液(食塩水) [KMスタック]
- 例4 液体詞 + 属性名詞
水が50gある。その体積は50ccである。
水は「体積」という属性をもつ。⇒「水の体積」
- 例5 液体詞 + 機能名詞
1本の試験管を用意し。その中に塩酸を入れる。
M-MFUNCHECK
「中」という意味を持つことのできる名詞(即ち「容器」)を上位概念にもつ名詞をスタック中よりさがしてくる。
⇒「試験管の中」
- 例6 液体詞 + サ変名詞
塩素酸ナトリウムを試験管に入れて熱する。この実験からわかることを述べよ。
前文へのリンクをつける。

Fig. 6 Examples of Analysis of *SONO* and *KONO*

要な要素が不足していれば、省略が行われたと考えられる。また、その単文・名詞句を構成する語をみることによってその省略語がどのような意味的内容を持つべき語かを知ることができる。

5.1 空白必須格の補足

単文があるままの意味内容を伝えるためには、その単文を支配する動詞の必須格要素が埋められていなければならない。従って、我々は単文内の係り受け関係が決定された時点で、動詞の格構造中でまだ埋められていない必須格要素があるかを調べ、もしあれば、前後の文からこれを埋めるべき語句を探索することにした。

動詞の格構造には、格要素になり得る名詞の上位概念が記述されているので、この下位概念になる名詞をM-スタック、KM-スタックから探索する。探索の順序は次の考え方に従って行なわれる。

- ① 4.2 で述べたと同様の理由から KM-スタック中の名詞を優先的に調べる。
- ② 前文の格要素になった名詞(名詞句の中心名詞)は他の名詞よりも優先される。(M-スタック中の

* 前3文よりも前の文の名詞でも、前3文中で(省略語、代名詞のreferentとして)補足された名詞は、3.1で述べたように、前3文中にスタックされている。

名詞アトムの順序がこれを表現している。)

③ 特に連用中止法で連続する文では、前文の格要素と同じものをそのままひきつぐのが普通であるので、空白必須格と同じ格で入っている前文中の名詞をまず調べる。

④ 探索は前3文*までを対象とし、それでも埋まらない格要素は後文を参照して埋める。(6.参照)

処理例を Fig. 7 に示す。

5.2 属性名詞・属性値名詞の処理

名詞は、それ自身で確定した意味を持つ entity word から、動詞、形容詞や英語における前置詞と同様に他の名詞と関係して初めて意味の定まる relational word に至るまで、連続的に分布している。我々はこのことを考慮して名詞を8個のカテゴリに分類した⁵⁾。ここで機能名詞や形式名詞のように relational word 的側面の強い名詞は、それ自身単独で名詞句を構成することはないが、属性名詞・属性値名詞は entity word と relational word の中間的性質を持ち、文構造上はそれ自身で名詞句を形成しながら、意味的には、はなれた位置にある物質

名詞と呼応している場合がある。この例を Fig. 8 に示す。

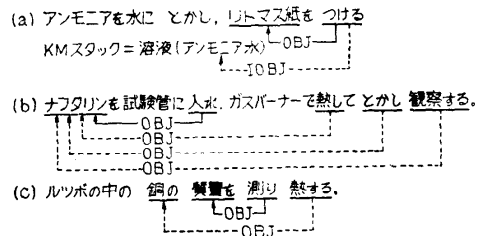


Fig. 7 Examples of Filling up Blank Case-Elements

- | | |
|------------------------------------|---|
| (a) 入力文 | (b) (a)の解釈 |
| ① 銅を熱したときの色の变化 | ①' 銅を熱したときの(その銅の)色の变化 |
| ② 気体の温度を一定にしておいて、体積を増加させると圧力は減少する。 | ②' 気体の温度を一定にしておいて、(その気体の)体積を増加させると(その気体の)圧力は減少する。 |
| ③ 試験管に塩酸を20cc入れる。 | ③' 試験管に20ccの塩酸を入れる。 |
| ④ あふれた水をメスシリンダーにとり、体積を測ったら5ccあった。 | ④' あふれた水をメスシリンダーにとり、(その水の)体積を測ったら5ccの体積があった。 |

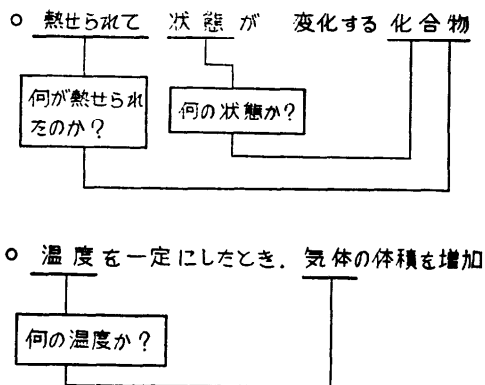
Fig. 8 Processing of Attribute-Nouns and Value-Nouns

そこで名詞句の解析が終了した時点で、対応する「物質名詞」を持たない属性名詞・属性値名詞が存在する場合、それらの名詞が関係する属性を持ち得る名詞を M-スタックから探し出す*。また埋込み文中に属性名詞・属性値名詞がある場合は、対応する名詞がその埋込み文よりも後にあることが多い。従って、この場合には次節で述べる後文の参照機能が必要になる。

5.3 後文の参照

文脈情報を用いる際には大抵の場合、前文のみを参照すれば良いことが多い。しかしながら Fig. 9 の様に前文には全く情報がなく、後文を見てはじめて正しい解析のできるものがある。前文はすでに解析されているため、その参照は比較的容易であるが、後文はまだ解析が行われていないため、参照の必要が生じた時点で即座にその操作に移ることはできない。従って、我々は E. Charniak¹⁾ の 'デモン' の考え方を採用し、処理中に後文参照の必要が生じた時点では、これを保留し、後文中のあるまつまり(名詞句)が解析されたときに、この保留された参照を行うメカニズムを、Fig. 10 に示すトラップリスト(trap list)の形で構成した。

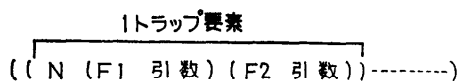
ここで1つのトラップ要素は1個の未解決な問題に対応しており、前文を参照しても解決しなかった場合に作成され、その時点のNは0である。そして、後の文の解析が進み、1つの名詞句の解析が終了すると、その名詞句の名詞をとり出し、トラップリスト中の各



させる時、その圧力は減少する。

Fig. 9 Examples that omitted words appear in succeeding sentences

* KM-スタック中の名詞はこのような省略はされず、普通「水素と酸素を混合し、この体積……」のように指示詞がつけられる。



N: 数字(名詞を読込七度に1ずつ増える)
F1: 後文参照用の関数
F2: F1の後始末用の関数

Fig. 10 Trap-List

トラップ要素の2番目の関数 F1 を順次実行してゆく。この関数 F1 としては

(a) 空白必須格を埋めるための関数

(b) 属性名詞に対応する物質名詞を見出す関数が LISP 関数として定義されている。これ以外に将来新たな機能が必要となれば、LISP 関数として定義することにより、トラップ要素に使用することができる。これらの関数は、現在出現した名詞が自分の捜していた目的に合うものであるかどうかをチェックし、

(i) もしも目的に合えば必要なデータの変更(例えば空白必須格の補足においては、対応する文の動詞の格構造に現在の名詞を付加する)を行い、関数の値として DELETE を返す。DELETE が返されると、システムはこの問題が解決されたと思なし、トラップリストからこのトラップ要素を除去する。

(ii) 解決しなかった場合、この問題は再び保留され、そのままトラップリストの中に残されて、Nが1だけ増す。

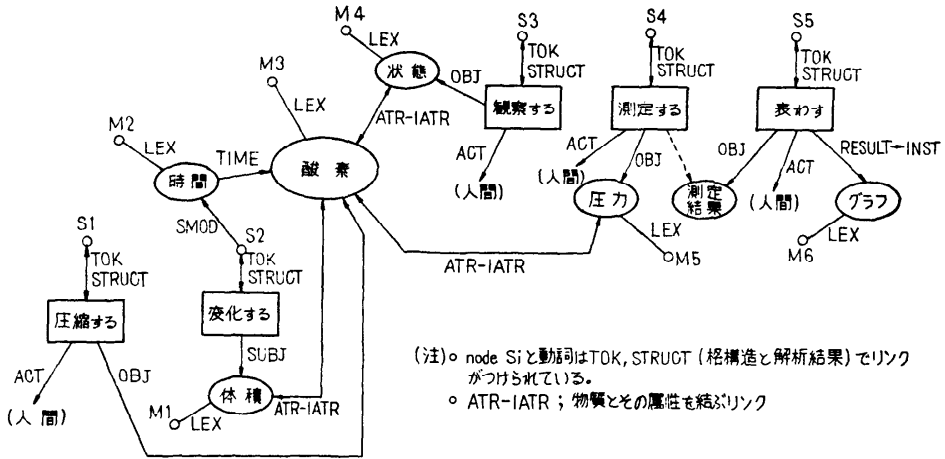
Nが4を越えた問題、即ち4個以上の名詞とのチェックを行い、それでも解が見つからなかったものは、自動的にトラップリストから消去される。ただし消去される前に関数 F2 が実行される。この F2 によって、解決されなかった問題の後始末や、必要な箇所へのバックトラッキングを行う。

このトラップリストの考え方は、主要な解析プログラムとは独立に、名詞が出現するたびに必要なプログラムが自動的に呼び出されて実行されるために、メインプログラムを複雑にすることなく、後文の参照が可能になっている。

6. 複雑な文の分析とその結果

本章では、前論文⁵⁾と本論文で述べた解析プログラムを複雑な文に対して適用した例を示し、これに従って、どの様に全体として働くかを述べる。

〈例1〉「圧縮されて体積が変化する時の酸素の状態を観察し、その圧力を測定し、それをグラフに表わ



(注) node Siと動詞はTOK, STRUCT (格構造と解析結果)でリンクがつけられている。
 ○ ATR-IATR ; 物質とその属性を結ぶリンク

(例1) 「圧縮されて体積が変化する時の酸素の状態を観察し、その圧力を測定し、それをグラフにする。」

Fig. 11 Example 1

す。」

処理ステップを以下に示す。

(1) 「圧縮され」の解析。この単文内では深層レベルの行為者格、目的格が規定されていない。「圧縮する」は人間を行為者格に要求するので、これはトラップ要素とはならず、目的格の探索がトラップ要素として設定される。

(2) 「体積が変化する時の酸素の状態」の解析。属性名詞「体積」に対応する物質名詞を探索するトラップ要素が設定される。

(3) (2)で解析された文は「連体形+形式名詞」の形式をとっているため、この埋込み文は無条件に形式名詞「時」を形容すると判断される。次に名詞句「時の酸素の状態」が処理され、この名詞句中の各名詞が埋込み文の格要素になり得るかどうかチェックされる。また、この名詞句の処理が終わった段階で、(1)、(2)で設定されたトラップ要素が解決され、「圧縮する」の目的格、「体積」に対応する物質名詞が、いずれも「酸素」と解釈される。

(4) 「……状態を観察し」の解析が行われる。

(5) 「その圧力を測定し」の解析。名詞句「その圧力」の解析では、まず限定指示とみて解析されるが失敗し、「圧力」を属性として持つ名詞として「酸素」が選出される(代行指示)。

(6) 「それをグラフに表わす」の解析。KM-スタックの探索によって、代名詞「それ」が「測定結果」を指示することを知る。また、深層格構造への変換規

則によって「グラフに表わす」が「グラフで表わす」と同じ解析結果に変換される。

解析結果の図示を Fig. 11 に、計算機からの出力の一部を Fig. 12 (次頁参照) に示す。

その他の処理例を Fig. 13 (次頁参照)、14、15 (p. 27 参照) に示す。

我々の作成した構文解析のプログラムで中学校理科化学の教科書および問題集中の文を解析した結果を

Table 3 Score of success and failure of the sentence analyses.

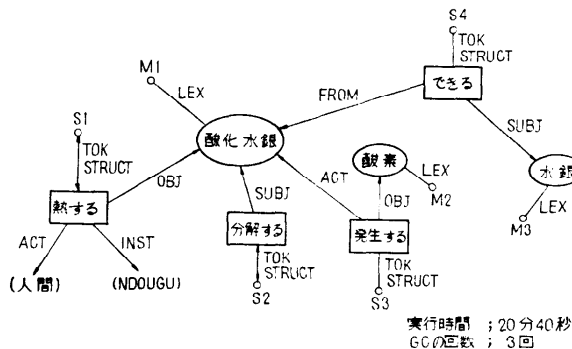
(1) 名詞句の処理 312 例中 成功 286 例 失敗 26 例	[失敗する例] ・ふつうの状態 ・化学変化以外の変化 ・原子の考え方 ・原子のモデル ・化学変化の方向 ・二酸化炭素へのとけ方 ・1cm ³ あたりの重さ ・試験管の下の部分
(2) 並列句の処理 372 例中 成功 349 例 失敗 23 例	[失敗する例] ・物質のかたさやとけ方 ・酸化銅の還元と定比例の法則 ・化学変化と物質の量 ・ピストンの重さ、筒とピストンのまさつおよび注射器内の温度変化
(3) 動詞文の処理 280 例中 成功 254 例 失敗 26 例	[失敗する例] ・大形注射器を2本用意して、一方には酸素を5cm ³ とり、他方には銅と希硝酸との作用でできる一酸化窒素を10cm ³ とる。 ・銅板、イオウの粉、変色した物質を、別々に3本の試験管にとり、それぞれに塩酸を加える。 ・いろいろな物質をつくっているものになるもの ・熱や光を認めにくいこともある。

```

M1
=LEX=
(TAISEKI 1)
=SATR=
((1 (IATR M3)))
=CASE=
((SUBJ. S2))
M2
=LEX=
(JIKAN 1)
=SATR=
((1 (TIME S2)))
=SMOD=
S2
M3
=LEX=
(SANSO 1)
=SATR=
((1 (ATSURYOKU M5) (STATE M4) (JIKAN M2)
(TAISEKI M1)))
=CASE=
(OBJ. S1)
:
S1
=TOK=
(D ASSHUKUSURU 2)
=STRUCT=
((((ACT (NINGEN M1)) (OBJ (KITAI M1)) (INST))))(3 0 0 1)
(OBJ. M3)) (*)
=MOD=
(UKEMI)
S2
=TOK=
(D HENKASURU 4)
=STRUCT=
((((SUBJ (RYOU M1)))) (1 1 0 0) ((SUBJ. M1)) (*))
((((SUBJ (BUSSHITSU M1)))) (1 0 0 1) ((SUBJ. M3)) (* (M
(TOKM M1) (TOK TAISEKI)) (AUXK (TOK GA) (CASE
(* (ACT SUBJ))))))
((((SUBJ (YOUSOU M1)))) (1 0 0 0) NIL (* (M(TOKM M1)
(TOK TAISEKI)) (AUXK (TOK GA) (CASE (* ACT
SUBJ))))))
=MOD=
NIL
:

```

Fig. 12 Computer Output of a Result



【例3】「酸化水銀を熱すると、分解して酸素を発生し、水銀ができる。」

Fig. 13 Example 2

Table 3 (前頁参照) に示す。実行時間は我々の研究室の小型計算機 (TOSBAC-40) の LISP で最大 28 分 30 秒 (GC* は 5 回, 含まれる単文数 6, 名詞 18 語), 平均 12~15 分 (GC は 2~3 回) であった。

失敗する特徴的な例を次にあげておく。

(a) 「グラフ用紙の横軸に亜鉛の質量を、縦軸に発生した水素の体積をとる。」

この例では、2つの文が並列しており、しかも前半の文の動詞が省略されている。我々の作成した解析プログラムは単文処理を基礎とし、まず述語を見出すことから処理が進められるため、省略された述語を補って解析することはできない。

(b) 「実験で求めたメチルアルコールが燃えるときに出る熱量を表8と比べてみよう」

我々の解析プログラムでは動詞連体形による埋込み文は、直後の名詞句内の名詞を形容するとして処理される。従ってこの例では、「実験で求めた」は「メチルアルコール」を形容するとみなされ、「求める」の目的格は「メチルアルコール」を採用する。しかしながら、ある文脈のもとでは、「実験で求めた」は「熱量」を形容し、「求める」の目的格を「熱量」とした方がよい場合もある。この様に、埋込み文が、間に文を介して、より後の名詞を形容する構造は現在処理できない。

(c) 「残った気体が酸素のときは実線、水素のときは点線で表わせ」

この例は(a)の様な単なる述語の省略とは異なり、「残った気体が酸素であるときは」という名詞文が変形をうけ、述語部分の根跡がなくなっており、単に述語を埋めるという操作では解決しない。

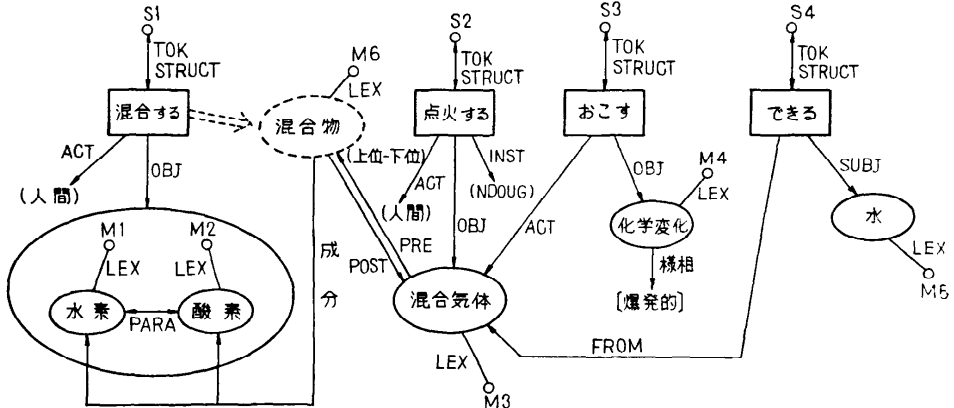
(d) 「銅板、イオウの粒、変色した物質を別々に3本の試験管にとり、それぞれに塩酸を加える。」

この文を解析するには、深層格構造とは別に、「銅板の入った試験管」、「イオウの入った試験管」等が存在することを示す内部表現を定め、これを用いて「それぞれ」という語を解釈する必要がある。

(e) 「水と食塩との混合比を 10 : 1 にする。この溶液を……」

現在のシステムでは、動詞を中心にインプリシットな知識を埋め込んでいるが、この例では動詞の介在なしに「溶液」が存在することを知

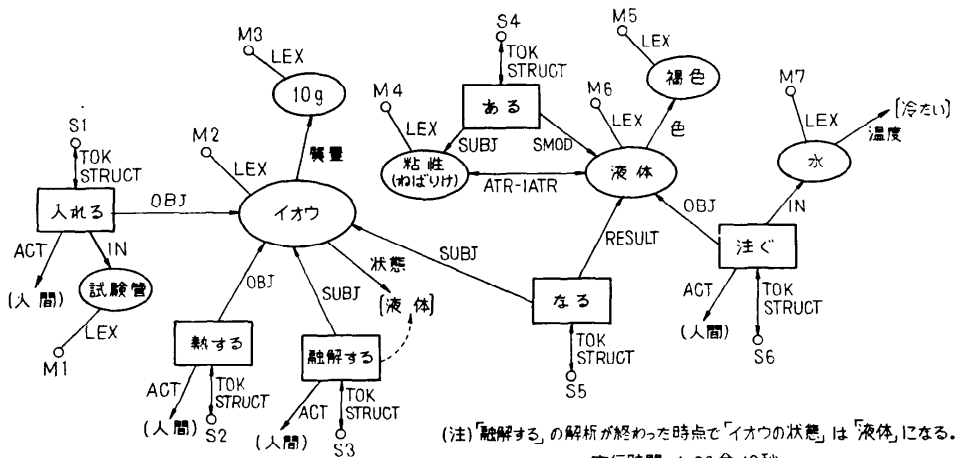
* GC: ガーベッジ・コレクション



実行時間；21分10秒
GCの回数；3回

【例3】「水素と酸素を混合し、この混合気体に点火すると爆発的な化学変化もおこし、水ができる。」

Fig. 14 Example 3



(注)「融解する」の解析が終わった時点で「イオウの状態」は「液体」になる。
実行時間；26分40秒
GCの回数；4回

【例4】「試験管にイオウを10g入れて熱し、融解して粘りけのある褐色の液体になったら冷たい水に注ぐ。」

Fig. 15 Example 4

る必要がある

なお名詞句処理の失敗は、係り受け関係はわかっても、解析後の内部構造をどうするかが決まらないものがほとんどである。

7. むすび

自然言語の大きな特徴は、表現の多様性と表現のあいまいさである。我々の解析プログラムでは同一の意味内容を表わす多様な表現を、できる限り吸収した深層表現におとすことを目指した。また、表現のあいま

いさは前後の文脈の参照によって解消できると考え、いままであまり手のつけられていなかった文脈処理を計算機的に実現した。アナホーラの解釈、省略語の推定はこの文脈処理によって可能になった。我々は、人間が文章を理解する際には、論理的操作を行う以前に、いわゆる短期記憶的なメカニズムで話題の中心となっているものを把握し、比較的単純な文脈処理を行っていると考えた。そして、これを計算機上を実現するために、M-スタック、KM-スタック、トラップリスト等の考え方を導入し、これらと単語の意味記述を結び

つけることにより、かなり有効な文脈処理を行い得ることを示した。

もちろん、人間の行っている文脈処理には、より論理的な側面がある。例えば、「試験管に食塩がある。この試験管に水を入れる。この溶液を……」では、「同一容器に入れられたものは、混合する。」という知識を使わないと解釈できない。これを実現するためには、我々のとった意味記述よりも、もっと広い範囲の実世界についての知識、及びより抽象化された「世界の記述」を必要とするであろう。しかし、入力された文章から直接整理された内部表現へ変換するよりも、入力文→表層格構造→深層格構造→短期記憶→長期記憶（状態記述）といったように、順次整理された構造へ変換し、浅いレベルで解決できない問題はより深いレベルの複雑な論理操作で解決してゆくのが得策であろう。

本システムは質問応答システム (Q. A. S) の入力文解析に使用する予定であるが、この Q. A. S の内部表現及び推論能力は、上で述べたような問題を解決する

のに役立つであろうと思われる。本研究の一部は文部省科学研究費補助金によった。

参考文献

- 1) Charniak, E.: 'Toward a Model of Children's Story Comprehension,' (MAC TR-266, 1972)
- 2) 林四郎:「指示連体詞「この」「その」の働きと前後関係」, (電子計算機による国語研究 IV, 国立国語研究所, 1972)
- 3) Winograd, T.: 'Procedures as a Representation for Data in a Computer Program for Understanding Natural Language,' (MAC TR-84, 1971)
- 4) 田辺英樹:「計算機による自然言語文の文脈解釈」, (京大特別研究報告書, 1975)
- 5) 長尾真, 辻井潤一, 田中一敏:「意味および文脈情報を用いた日本語文の解析一名詞句・単文の処理」, 情報処理, Vol. 17, No. 1, pp. 10-18 (1976)

(昭和50年4月28日受付)

(昭和50年7月7日再受付)