

TR-I-0039

解析用辞書開発作業に関する一考察
How to Make Correct Lexical Descriptions
Easily

逸見 研一
Ken'ichi Itsumi

1988.8

概要

解析用辞書の記述は、その妥当性を解析の試行により検証しながら進められる。この方法によれば、辞書記述上に問題がある場合、不首尾な結果が得られる事から問題の存在が判明するだけで、どこをどう直せばよいのか知るためには、プログラムのデバッグのように、トレース情報などから、語彙記述のレベルに還元する操作が必要となってくる。この作業は一般に簡単ではなく、この点が解析用辞書開発作業のボトルネックとなっている。本稿では、語彙記述の機能からその記述が妥当であるための必要十分条件を計算する方法を述べ、さらに、こうした条件下での記述を容易に行う方法を明らかにした。このように妥当な範囲で辞書記述を進めることによって、処理の試行による検証作業に代えることができ、上述の困難を避けることが期待できる。

ATR Interpreting Telephony Research Laboratories
ATR 自動翻訳電話研究所

1. はじめに

機械翻訳システムなどの自然言語処理システムに必要な言語情報、すなわち、一般的な文法情報や語彙情報などは、一般に、その妥当性の検証を行いながら蓄積・整備される。従って、この過程を効率的に行うことが、このようなシステムを構築する上での重要な要素となる。

ある語彙に関する記述の妥当性を評価しようとする場合、最も直接的な方法は、実際にその記述を用いて、目的とする処理を試行して見ることである。これにより直ちに判ることは、満足な結果が得られなかった場合には、その処理の中に何か不適切な記述が存在するということであり、具体的な対処法—どの記述をどのように修正すればよいのか—ということ—は、直接には見えてこない。そこで、対処法を見出すために、ちょうどプログラムのデバッグと同じ様に、問題点を、トレース情報などから、語彙記述のレベルにまで還元することが必要となってくる。だが、これは容易な作業ではなく、この点が機械辞書開発作業の主要なボトルネックとなっている。

ところで、処理の試行によって、「この語彙記述は妥当か?」が判明するわけだが、辞書開発者にとって、本当の問題は、「この語彙記述は妥当か?」ではなくて、「どのような語彙記述が妥当なのか?」であるはずである。そして、「どのような語彙記述が妥当なのか?」から始めること、すなわち、あらかじめ記述が妥当であるための条件を知ったうえで、それを遵守しながら妥当な範囲のなかで辞書記述を進めることによって、上述の「この語彙記述は妥当か?」式チェックと、それに伴う困難を避けることができ、より見通しよく効率的に辞書記述作業を進めることが期待できる。

本稿では、ATRで開発されている端末間対話翻訳システムNADINEにおける解析用辞書の開発作業を対象として、極めて限定的にはあるが「どのような語彙記述が妥当なのか?」を明らかにし、その制約のもとで、辞書記述を逐行する手法の原理について述べる。

2. 単一化パーサー

後に続く説明のために、端末間対話翻訳システムNADINEおよびその単一化パーサーについて概略を述べる。

2.1 NADINEの単一化パーサー

ATRで開発している端末間対話翻訳システムNADINEは、自動翻訳電話のプロトタイプとしての、端末間でのキーボードを介しての日英二カ国語間の対話のための機械翻訳システムである。このシステムの日本語解析部は、単一化文法の枠組みに基づいている。

NADINEの日本語解析部は、人力として、解析の対象である文字列と、単一化文法の枠組みに従って記述された文法規則及び語彙記述をとり、Earleyのアルゴリズムに従って解析を逐行し、統語的・意味的に受理された構造各々の素性構造表現の集合を返す。文法的な枠組みは、語彙中心的で、Head-Driven Phrase Structure Grammar [HPSG] の枠組みに沿っている。そして、その日本語版である Japanese Phrase Structure Grammar [JPSG] の主要な素性を採用し、これに加え、話し言葉を記述するために、文の階層性を示す素性や語用論的な素性などを新たに導入している

2.2 単一化文法の利点

単一化文法の枠組みを用いることの利点として、次の2つがある。

- (1) ある語の担う統語論的・意味論的・語用論的情報などの多面的な情報を、素性構造の形式を用いて統一的に記述することができる。
- (2) 自然言語の解析、変換、生成、などの多様な処理を、素性構造間の単一化演算という論理的に明解な形式で構成することが出来る。

こうした利点によって、単一化文法の枠組みに従うことにより、高度で複雑な処理を、比較的見通しよく行うことが期待できる。

2.3 単一化文法の枠組みでの語彙記述

単一化文法の枠組みでは、文や名詞句などの句構造、さらにそれを構成する各単語の担う情報は、いわゆる素性の束である素性構造で表現される。[Fig.1]に、NADINEの解析用辞書における記法に従った「情報」という語彙項目の記述例を示す。

2.4 単一化文法の枠組みでの文法規則の表現

単一化文法の枠組みでは、文法規則は、規則左辺の記号の構成を表現するCFG規則と、そのCFG規則両辺の記号間に満たされるべき制約によって表

```
(deflex 情報 N
  [[head [[pos n]]]
   [subcat {}]
   [slash {}]
   [semf [[anim -]]]
   [sem [[parm ?x]
         [reln 情報-1]
         [obje ?x]]]])
```

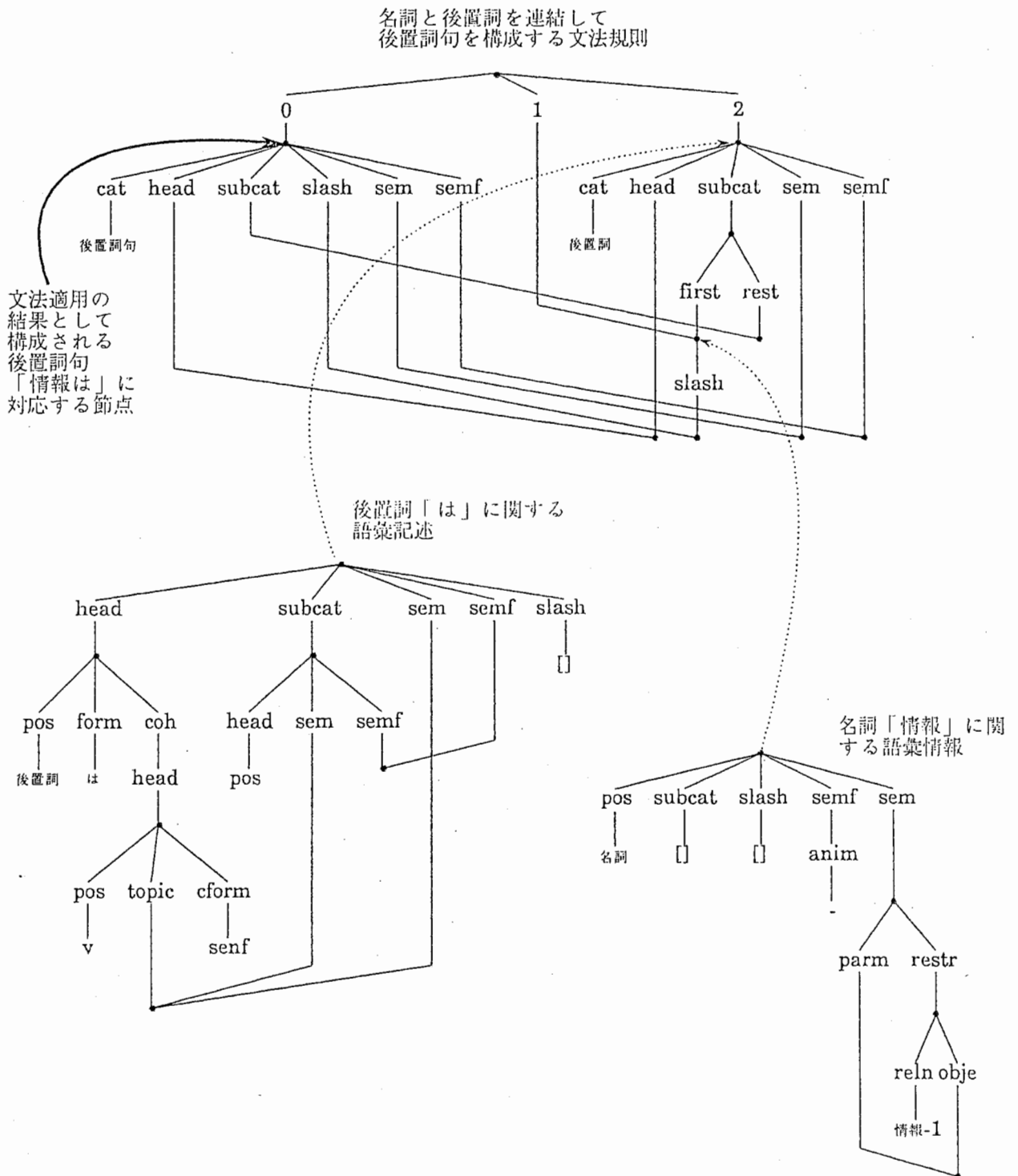
[Fig.1 語彙項目に関する記述の例]

```
(defrule 後置詞句 => (名詞 後置詞)
  (<0 head> == <2 head>)
  (<0 subcat> == <2 subcat rest>)
  (<0 slash> == <1 slash>)
  (<1> == <2 subcat first>)
  (<0 sem> == <2 sem>)
  (<0 semf> == <2 semf>))
```

[Fig.2 文法記述の例]

第1行が後置詞句→名詞 後置詞なるCFG規則を表し、後に続く等式の列は、規則が適用されるとき、規則両辺の記号間に満たされるべき制約を表す。

される。[Fig.2]に、NADINEの解析用文法の記法



[Fig.3 語彙中心の単一化文法の枠組みでの文法規則の適用]

に従った、名詞と後置詞の連結から後置詞句を構成する文法規則の例を示す。

第一行defruleの後に続くのがCFG規則で、これは、矢印左辺の記号が、矢印右辺の記号間の連結によって構成されることを示す。

この後に続く次に抜粋したような、素性構造のパス表現間またはパス表現とアトム間の等式によって、規則が適用される際に、CFG規則両辺の記号間に満たされるべき制約が記述される。

(<0 head> == <2 head>)

パス表現によって素性構造中の任意の素性の値を参照することができる。パス表現は、非負の整数とそれに続く素性名の列挙により構成される。非負の整数は、0ならば規則左辺の記号、自然数nの時は、規則右辺の、左からn番目の記号に対応する素性構造を示し、素性名の列挙は、その素性構造の根から、参照する素性にいたるまでの素性名を示す。パス表現間の等式によっては、その両辺に対応する二素性が同じ値を共有しなければならないという制約が示され、パス表現とアトムの等式によっては、そのパス表現で指示される素性の値がそのアトムでなければならないという制約をあらわす。

例えば、上に挙げた例では、このCFG規則によって後置詞句が構成される時、出来上がる後置詞句のhead素性の値は、後置詞のhead素性の値と等しくなければならないという制約を表している。

こうした文法適用時の制約もまた素性構造で表現することができる。[Fig.2]にあげた文法規則は、[Fig.3の上部]に示した素性構造と同値である。

2.5 文法規則の適用

単一化文法の枠組みの上では、文法規則の適用は、規則右辺の記号に対応する素性の値それぞれに対して、対応する素性構造を単一化していくことにより逐行される。このとき単一化の順番は任意でよい。もし、途中で、こうした単一化に失敗したならば、その文法規則の適用は失敗ということになる。規則右辺のすべての記号に対応する素性の値に対して、この単一化の操作が無事終了すれば、この文法適用は成功し、このときの、規則右辺の記号に対応する素性の値が、文法適用の結果構成されたその記号に対応する素性構造となる。例として、[Fig.1]に挙げたの名詞「情報」と後置詞「は」から、[Fig.2]の文法規則に従う連結によって、後置詞句がつけられる例を[Fig.3]に示す。

3. 解析用辞書の開発作業の流れ

現在、我々のNADINEシステムでは、想定入力文ごとに、[Fig.4]に示したような手順を踏んで、解析用辞書の開発作業を行っている。以下主要なプロセスについて説明を加えておく。

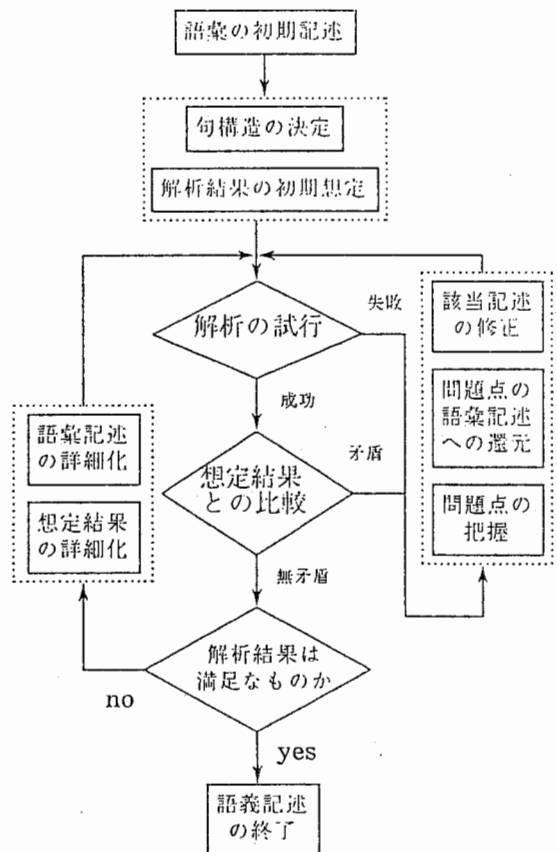
句構造の決定

想定入力文の句構造を決定する。この作業は視察により容易におこなうことができる。

解析結果の初期想定

もとの想定入力文と、選んだ句構造から、解析結果として望ましい素性構造を想定する。

普通、はじめから見通しのきくものではないので、はじめは句構造を反映したラフスケッチのようなもので、後に述べる解析の試行の結果を受けて、徐々に詳細なものにしていく。



[Fig.4 語彙記述開発作業の流れ]

語彙の初期記述

想定入力文を構成する単語[用言の場合には語幹と語尾]それぞれに対して、語彙記述を準備する。

語彙記述は、普通、類縁の語の語彙記述のコピーを適宜変更することによって作られる。これも、試行の結果を受けて、望ましいものとしていく

構文解析の試行

当面の語彙記述の妥当性を検証するために、その記述を用いて、想定入力文の解析を実際に行ってみる。

解析が途中で中座して失敗するならば、その原因を発見し、それを語彙記述のレベルまで還元して適当な修正を加える。

試行解析結果と想定解析結果との比較

試行解析結果と想定解析結果を比較し、二者間に矛盾があれば適当な修正を施す。

4. 語彙記述作業の問題点

我々のNADINEシステムの日本語解析部の解析用辞書の開発作業は、前章でみた様に、解析の試行による記述の妥当性の検証を中心に進められている。

作業を開始した段階では、目標とする解析結果や各単語の語彙記述は、十分に明確なものではないことが多いので、構文解析処理の試行によって、それら各々に関して、妥当性の検証を行い、その結果不備が発見されれば、それを補うことによって、各単語の語彙記述をより完全なものにしていくわけである。ここでは、語彙記述作業における処理の試行は、辞書記述の十分性を確認するための手段としてばかりではなく、それぞれの語彙記述に関して、それが妥当であるための条件の探索の道具として機能している。

だが、解析の試行によっては、語彙記述に何か不備があるときに、構文解析の失敗や、得られた解析結果の異常から、問題点の存在を知ることができるだけであり、直接問題解決につながる情報、すなわち、語彙記述のどこに問題があって、それをどう直せばよいか、などといった肝心のことがすぐには判らない。そこで、プログラムをデバッグするときのように問題点を特定の語彙記述に選

元する作業などが必要になってくる。これは、一般には容易ではなく、語彙記述の妥当性を解析の試行に頼る戦略においては、このプロセスがボトルネックとなっている。

こうした問題は、たかだか語彙記述の十分性を確認する機能しか有していない処理の試行という手段で、語彙記述に関する必要条件の探索おもしろいことの無理に起因するものと思われる。そして、処理対象文を構成する各単語について、その語彙記述に関する必要条件を明らかにする道具などを用意することができれば、このような困難を軽減することが期待できる。

5. 辞書記述の妥当性I

すでに述べたように、NADINEシステムの日本語解析部の解析用辞書の開発作業では、個々の想定入力文各々に対して別個に辞書が用意される。だからここでは、語彙記述の妥当性ということをして、特定の文の成分としての語義の妥当性に限定して議論を進めることにする。だから当面のところ、以下にあげる2種類の妥当性を考えていくことにする。

[妥当性5-1] 処理遂行に関する妥当性

意図した句構造に沿って解析を遂行しうること

[妥当性5-2] 結果に関する妥当性

解析結果中の素性の値を意図した値とすること

以下では、この二つの妥当性を満たすのに必要あるいは十分な条件を、文法規則と句構造から計算するための原理について述べることにする。

これ以外の妥当性については、改めて[8章]で議論する。

6. 文法適用演算の再検討

処理全体を通じての妥当性を論じるまえに、その準備として、個々の文法適用のステップにおいて、これが、成功するための条件、さらにこれが、意図した特定の結果をもたらすための条件などに注意して文法適用演算を眺めてみることにする。

6.1 記法の導入

議論のためにいくつかの記法を導入する。

素性構造間のユニフィケーション演算の単位元(即ち、空の素性構造)を、

$$[]$$

と表記することにする。また、素性構造 x_0, x_1, x_2, \dots 間のユニフィケーションを

$$\text{unify}(x_0 \ x_1 \ x_2 \dots)$$

とかくことにする。

いま、文法規則 X に関して、その素性構造表現を

$$R_X$$

と表記することにする。文法規則の素性構造表現は、[Fig.3]に示したように、CFG規則両辺にある記号各々に対応する素性を有している。

文法規則 X のCFG規則中の記号を左から順に X_0, X_1, \dots としたとき、 X_n に対応する R_X 中の素性の値を、次のように記すようにし、この操作を R_X を X_n に関して解くと呼ぶことにする。

$$X_n(R_X)$$

R_X 中の記号 $X_i (i=0, 1, \dots, n)$ に対応する素性構造に対して、素性構造 x_i を単一化したものを

$$\text{combine}(R_X \ x_0 \ x_1 \ \dots \ x_n)$$

単語 Z の語彙記述に対応する素性構造を

$$L(Z)$$

二つの素性構造 A, B がその部分構造として共有する素性構造の集合を

$$\text{積}(A \ B)$$

と表記することにする。また、二つの集合 S_1, S_2 の差集合のことを

$$S_1 \setminus S_2$$

と表記することにする。これに従うと、解析に際しての文法適用に伴う[Fig.3]に示したような演算、すなわち、文法規則の素性構造表現 R_X の右辺の記号 $X_i (1 \leq i \leq n)$ に対応する素性値に対して素性構造 x_i の単一化を行った後に、規則左辺の記号に対応する素性の値を求めることを下式のように表現することができる。

$$X_0(\text{combine}(R_X \ [] \ x_1 \ \dots \ x_n))$$

6.2 文法適用に伴う演算のVenn図表現

説明のために、文法規則の素性構造表現として、規則両辺の記号 X_i に対応する節点以下のノードを、 i を付した円のなかに描いたVenn図表現を導入する。以下簡単のために $i=0, 1, 2$ とするが、 i を任意の自然数としても、同様の議論を行うことができる。これに従えば、通常の文法適用演算は[Fig.5]のようになる。

ここで、あとに続く議論のために、さらに次の記法を導入する。

規則両辺の記号 X_j, X_k それぞれに対応する素性構造間で共有される素性構造の集合、すなわち[Fig.5]のようなVenn図表現では、 j を付された円と、 k を付された円の交わりにある素性構造にあたる下式を、

$$\text{積}(X_j(\text{combine}(R_X \ \dots)) \\ X_k(\text{combine}(R_X \ \dots)))$$

次のように略記することにする。

$$X_j \cap X_k(\text{combine}(R_X \ \dots))$$

この式の値は、素性構造の集合である。また、

$$X_j \cap X_k(\text{combine}(R_X \ \dots)) \setminus X_j \cap X_k(R_X)$$

のことを、

$$X_j \cap X_k(\text{combine}(R_X \ \dots) \wedge R_X)$$

と略記することにする。

また、記号 X_j に対応する節点に支配され、かつ記号 X_k に対応する節点に支配されない構造に、両者が分かち持つ構造($X_j \cap X_k(\text{combine}(R_X \ \dots))$ の要素)の根をくわえたものを、

$$X_j - X_k(\text{combine}(R_X \ \dots))$$

と表記することにする。[Fig.6参照]

6.3 文法適用結果の構成

[Fig.5]をみれば、文法適用結果は

$$X_0 - X_1 - X_2(R_X) \text{-----式5.2.1}$$

$$X_0 \cap X_1(\text{combine}(R_X \ [] \ x_1 \ x_2) \wedge R_X) \text{-----式5.2.2}$$

$$X_0 \cap X_2(\text{combine}(R_X \ [] \ x_1 \ x_2) \wedge R_X) \text{-----式5.2.3}$$

の和によって構成されているのが判る。ここで、規則Xの適用結果は、規則自身のみによって決定される規則右辺の値によらない項(式5.2.1)に対して、右辺記号の値により変化する、後の二項(式5.2.2)、(式5.2.3)の要素である構造を付加したものとみなすことができる。また、規則Xの適用が成功したとき得られる素性構造は常に、 $X_0(R_X)$ の拡張であることがわかる。

6.4 文法適用の逆演算

解析に際しての文法適用演算に対して、その逆の演算を考えることができる。いま、文法規則Xに対してこの逆演算を考えることにする。[Fig.5] にならって、文法適用の逆演算の過程をVenn図で表すと[Fig.7]のようになる。

6.5 文法適用結果の満たすべき条件

後に続く議論のために、素性構造xが文法規則Xの適用結果と成りうるための条件について議論しておくことにする。

[5.3節]の議論より、まず素性構造xが文法規則Xの適用結果であるためには、xは $X_0(R_X)$ の拡張でなければならない。これは、下式のように表現できる。

$$\text{unify}(x, X_0(R_X)) = x$$

この条件のもとでは常に、

$$\text{combine}(R_X, x, [], [])$$

を失敗することなく計算することができる。

また、この拡張とは $X_0 \cap X_1(R_X)$ および $X_0 \cap X_2(R_X)$ の要素を構成する節点の値が変化するものであるから、そうでない節点が文法適用の結果として値を変えることはないので、

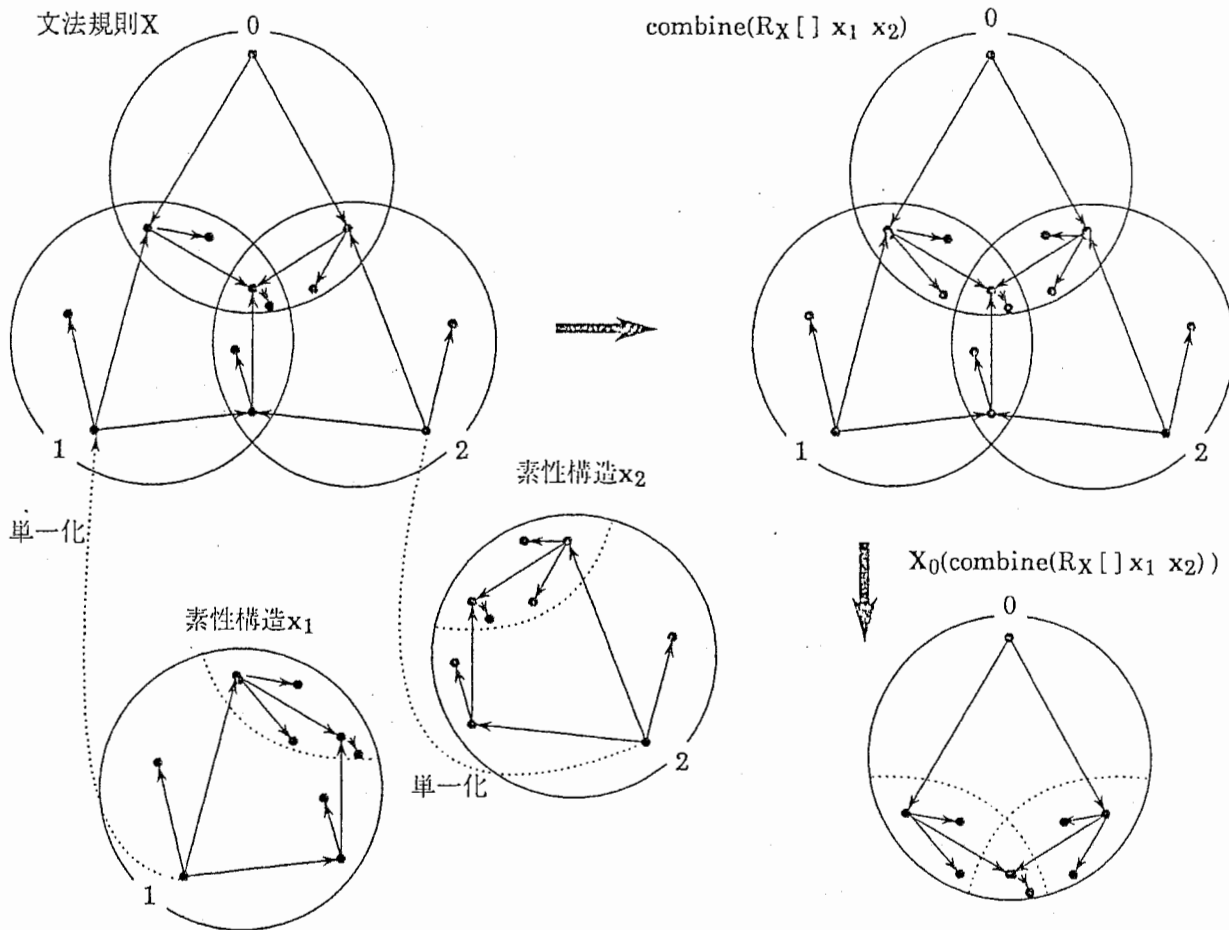
$$X_0 - X_1 - X_2(R_X) = X_0 - X_1 - X_2(\text{combine}(R_X, x, [], []))$$

でなければならない。

6.6 文法適用が成功するための条件

まず、文法規則Xの適用が成功するための、規則右辺の記号 X_1, X_2 に対応する素性構造 x_1, x_2 の満たすべき条件について議論しておく。文法の適用が成功する、すなわち

$$X_0(\text{combine}(R_X, [], x_1, x_2))$$



[Fig.5 文法規則のVenn図表現]

を失敗することなく計算するためには、これを構成する下に示す二つの単一化演算が成功しなければならない。

$\text{unify}(X_2(\text{combine}(R_X[] x_1 x_2) x_1) \neq []$
 $< x_2$ が確定していない、もしくは、
 $X_1 \cap X_2(\text{combine}(R_X)) = []$
 ならば
 $\text{unify}(X_1(R_X) x_1) \neq [] >$

$\text{unify}(X_2(\text{combine}(R_X[] x_1 x_2) x_2) \neq []$
 $< x_1$ が確定していない、もしくは、
 $X_1 \cap X_2(\text{combine}(R_X)) = []$
 ならば
 $\text{unify}(X_2(R_X) x_2) \neq [] >$

またこのとき、必ず文法規則Xの適用は成功する。

6.7 特定の文法適用結果を得るための条件

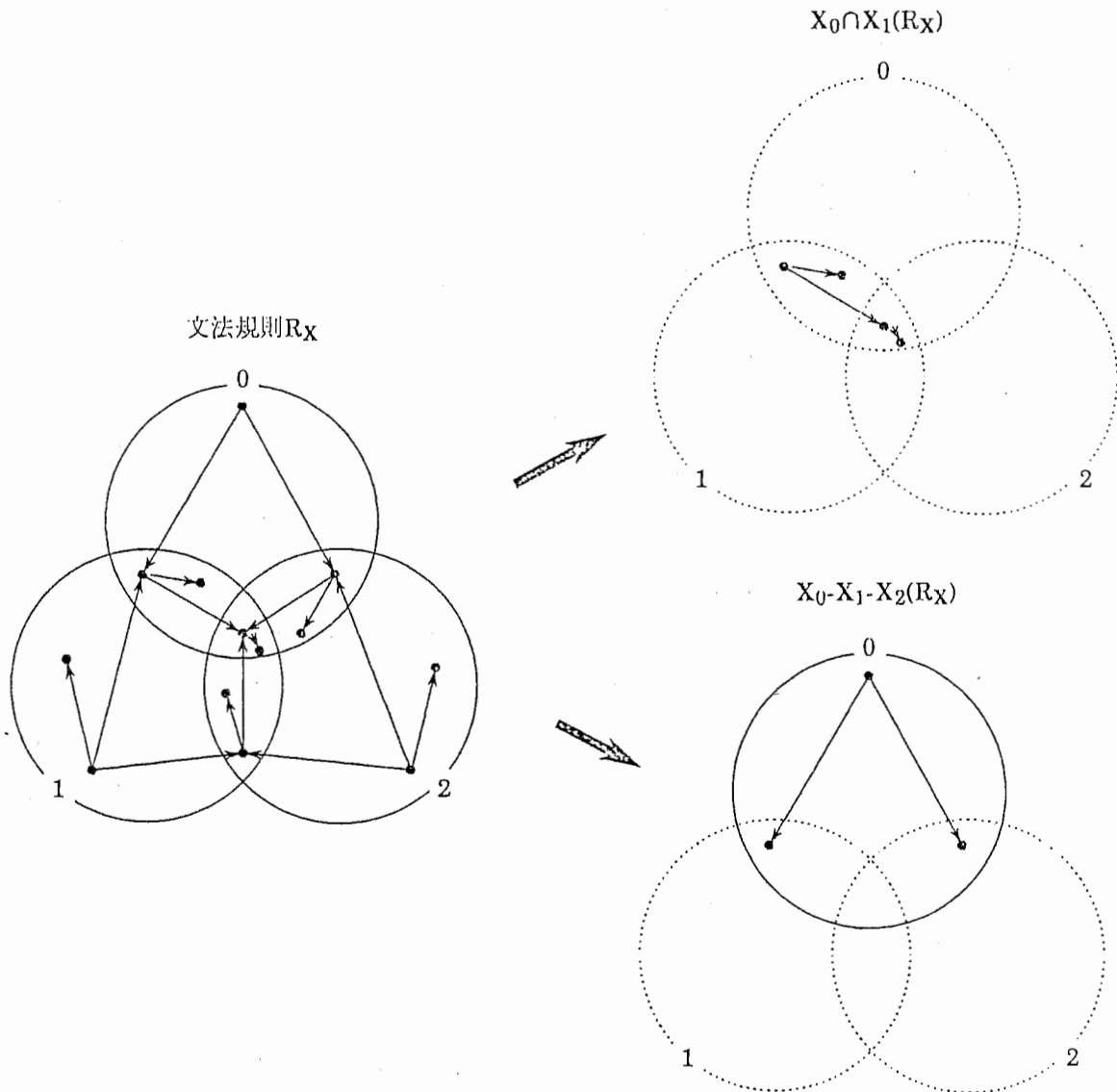
次に、文法規則Xの適用結果を、[6.5節]の議論により明らかにした条件を満たす素性構造xとするために、規則右辺の記号 $X_1 X_2$ に対応する素性構造 $x_1 x_2$ の満たすべき条件について考えてみることにする。

[5.5節]の議論より、素性構造xは、次の条件を満たす。

$$\text{unify}(x X_0(R_X)) = x$$

$$X_0 - X_1 - X_2(R_X) = X_0 - X_1 - X_2(\text{combine}(R_X x [[]]))$$

さらに、このためには、 x_1, x_2 に対する文法規則Xの適用が成功する必要があるから、[6.6節]の議論より、



[Fig.6 $X_j \cap X_k$ および $X_j - X_k$ の意味]

$\text{unify}(X_1(\text{combine}(R_X [] x_1 x_2) x_1)) \neq []$
 $\langle x_2 \text{が確定していない、もしくは、}$
 $X_1 \cap X_2(\text{combine}(R_X)) = []$
 ならば
 $\text{unify}(X_1(R_X) x_1) \neq [] \rangle$

かつ、

$\text{unify}(X_2(\text{combine}(R_X [] x_1 x_2) x_2)) \neq []$
 $\langle x_1 \text{が確定していない、もしくは、}$
 $X_1 \cap X_2(\text{combine}(R_X)) = []$
 ならば
 $\text{unify}(X_2(R_X) x_2) \neq [] \rangle$

であることが必要で、さらにこうして得られた結果が、意図した結果 x であるためには、

$$\begin{aligned}
 & X_0 \cap X_1(\text{combine}(R_X [] x_1 x_2) \wedge R_X) \\
 &= X_0 \cap X_1(\text{combine}(R_X x [] []) \wedge R_X)
 \end{aligned}$$

かつ、

$$\begin{aligned}
 & X_0 \cap X_2(\text{combine}(R_X [] x_1 x_2) \wedge R_X) \\
 &= X_0 \cap X_2(\text{combine}(R_X x_0 [] []) \wedge R_X)
 \end{aligned}$$

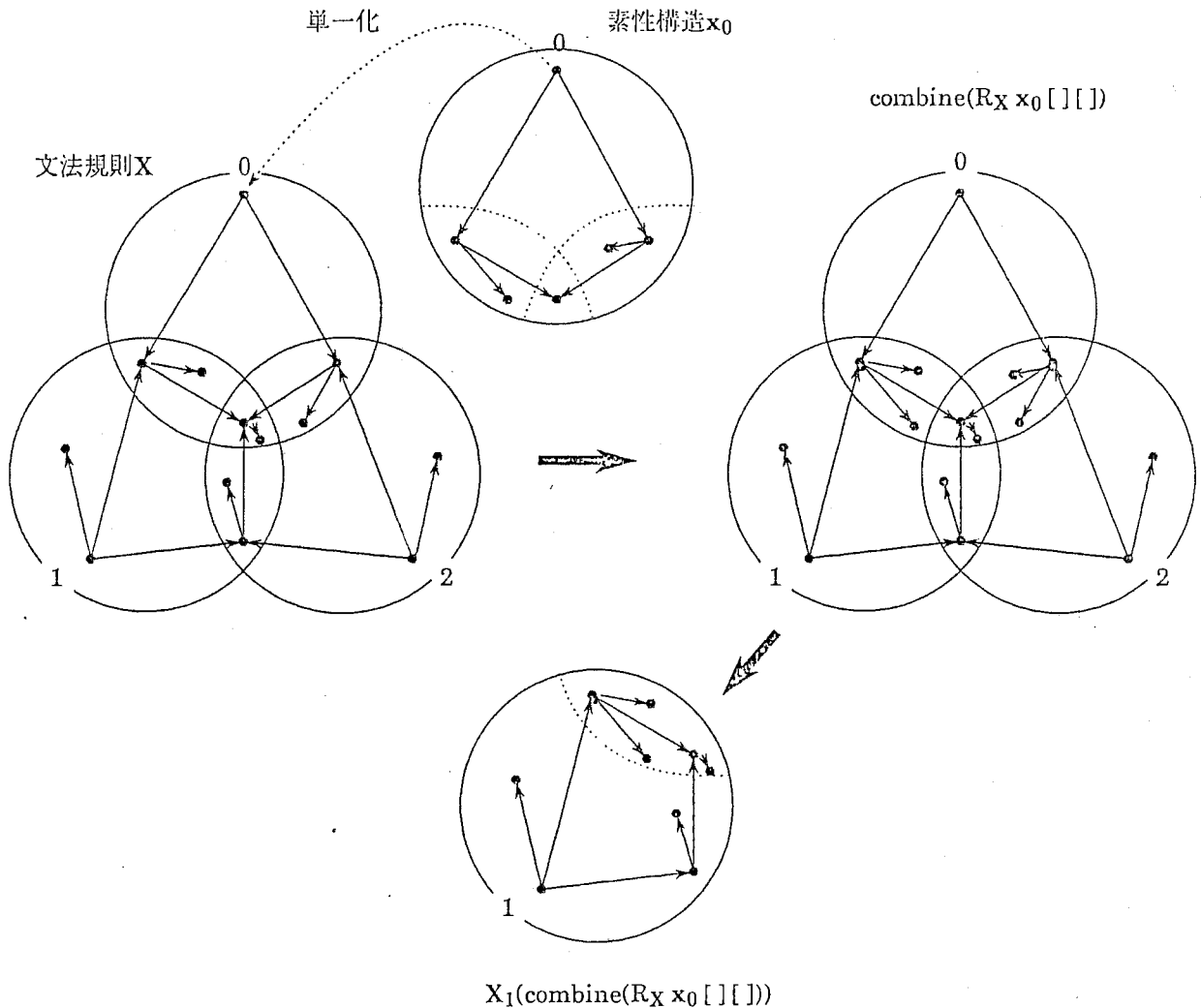
でなければならない、また、このとき得られる結果は必ず x である。

7. 構文解析演算の再検討

本章では、前章での、個々の文法適用に関して、その両辺記号の値の妥当性に関する議論の結果をもちいて、処理全体について同様の議論を試みることにする。

7.1 句構造と解析結果

ここでは、「情報はエントロピーだ」という文について考えることにするが、一般の文について同様の議論を行うことができる。必要な文法はす



[Fig.7 解析時の文法適用演算の逆]

で用意されているものとし、件の文を構成する単語各々に関する語彙記述は未知であるものとする。

この文を[Fig.9]の句構造Pにしたがって構文解析する処理は、前章で導入した記法に従うと次のように描くことができる。

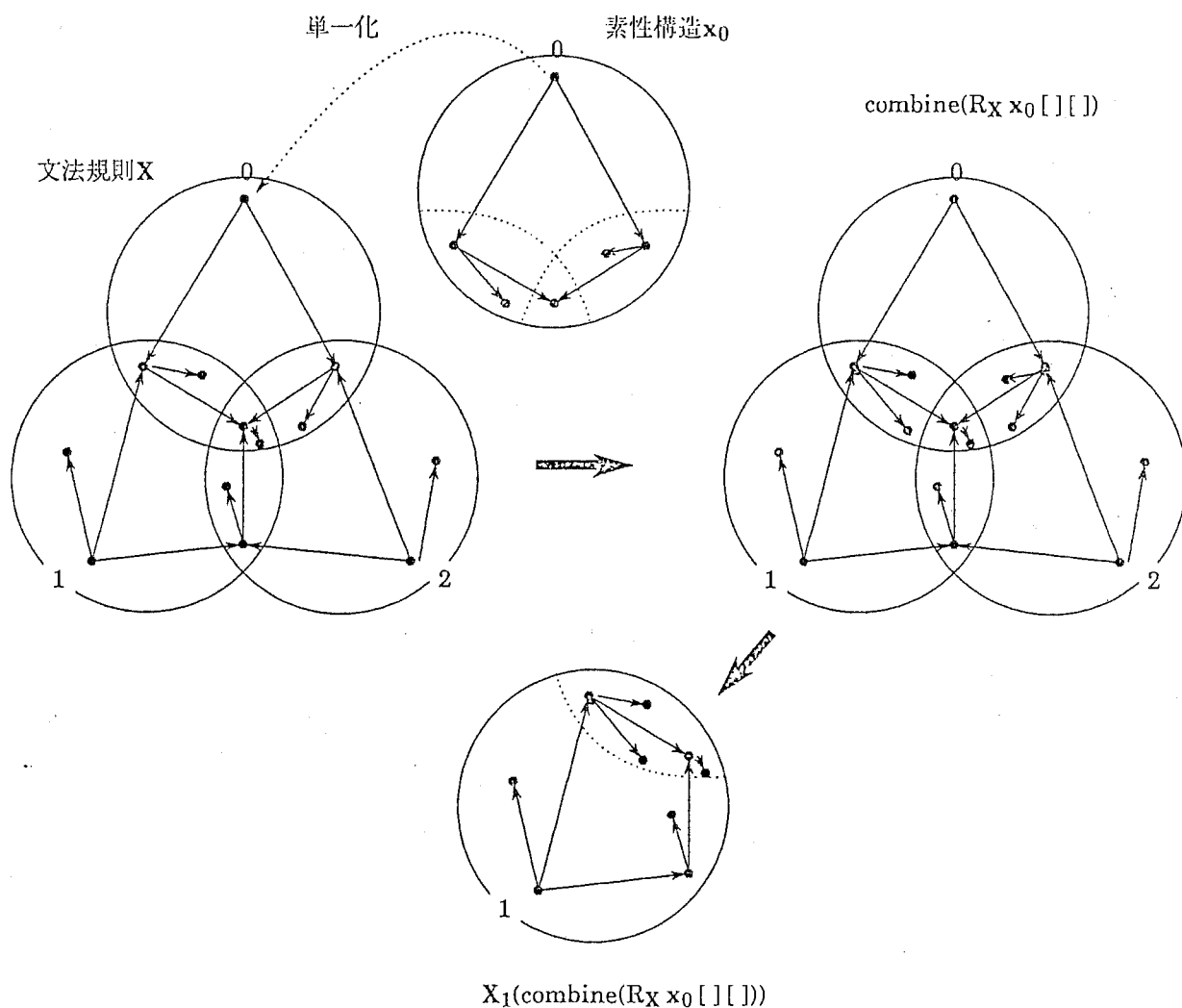
- 文法規則1 動詞句1 ⇒ 後置詞句 動詞句2
- 文法規則2 後置詞句 ⇒ 名詞 後置詞
- 文法規則3 動詞句 ⇒ 名詞 助動詞

動詞句1(combine(R₁ []
 後置詞句(combine(R₂ []
 L(情報)
 L(は)))
 動詞句(combine(R₃ []
 L(エントロピー)
 L(だ))))--(式7.1)

[なお、規則1のように、規則両辺に同じ記号がくる場合は、末尾に左から順に番号を付け弁別することにする。]

視察から、たとえば[Fig.9]のように、直ちに句構造が決定される。

仮定により、語彙記述は未知であるから、それらを[]とおくと下式のようになり、



[Fig.7 ある結果を得るために左辺記号の値に課せられる制約]

動詞句1(combine(R_1 []

後置詞句(R_2)
動詞句(R_3))----- (式7.2)

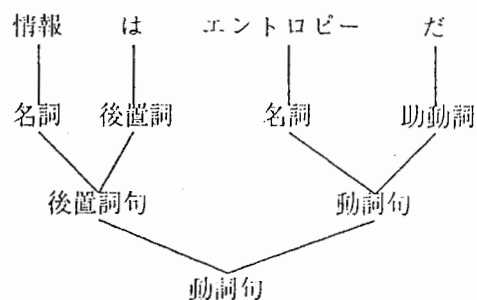
これは未知の項を含まないので、語義の如何を問わず計算することができる。この式の値の意味するところについて検討してみよう。[6.5節]の議論より、文法規則3の適用成功時の結果は常に動詞句(R_3)の拡張であり、文法規則2の適用成功時の結果も同様に後置詞句(R_2)の拡張であることから、件の句構造に従った構文解析の成功時の結果は、(式7.2)の値の拡張であることがわかる。(式7.2)中の項

combine(R_1 []
後置詞句(R_2)
動詞句(R_3))----- (式7.3)

を、もとの句構造の句構造条件と呼び、句構造Pの句構造条件を

C(P)----- (式7.4)

と表記することにする。(式7.2)は(式7.4)より、



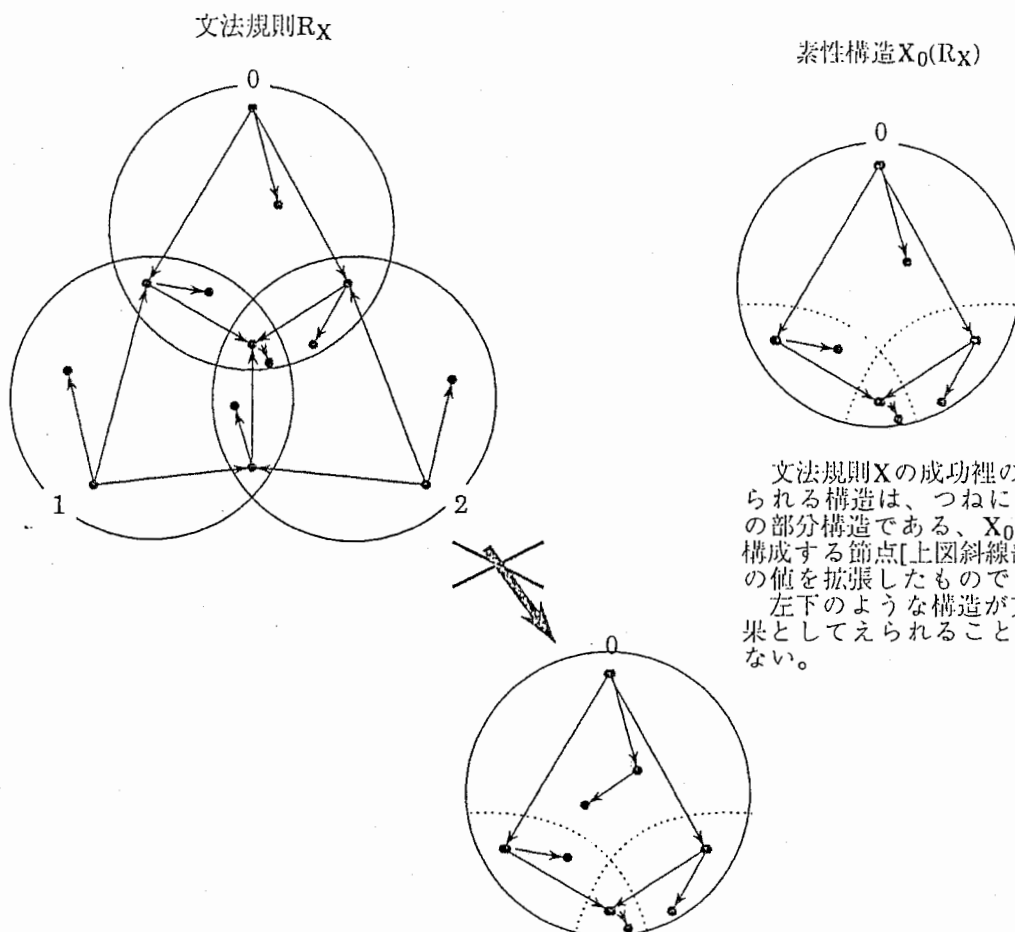
[Fig.9 「情報はエントロピーだ」の句構造]

動詞句1(C(P))----- (式7.5)

と表記することができる。これを解析結果に関する句構造条件と呼ぶことにする。(式7.5)は、最小句構造条件を開始記号に関して解く、と読むことにする。

7.2 構文解析が成功するための条件I

意図した句構造に従って、文の解析を逐行するためには被解析文を構成する各単語の語彙記述が、



文法規則Xの成功裡の適用結果として得られる構造は、つねに、素性構造 $X_0(R_X)$ の部分構造である、 $X_0 \cap X_1(R_X)$ の要素を構成する節点[上図斜線部に含まれる節点]の値を拡張したものである。
左下のような構造が文法規則Xの適用結果としてえられることはけっしてありえない。

[Fig.8 文法規則Xの適用が成功した場合にその結果として考えられる素性構造の満たすべき条件]

の議論により、句構造が開始記号の値に課する制限を知ることができる。

7.4 ある結果を得るための終端記号の値の制約

さて、今度は、[7.3節]に述べたような句構造の開始記号と成りうるある値 x を開始記号の値とするために、各終端記号の値に課せられる制約のことを考えることにする[Fig.13参照]。これと同様の議論によって、ある結果を得るために、各終端記号に課せられた制約を知ることができる。

7.5 視覚化による語彙記述の支援

以上に述べてきた、語彙記述が、意図する句構造の適当な終端記号となるための条件、ならびに、ある結果を実現するために各終端記号の満たすべ

き制約は、いずれも、ある素性構造と単一化可能であるという形で求められることに注意して欲しい。

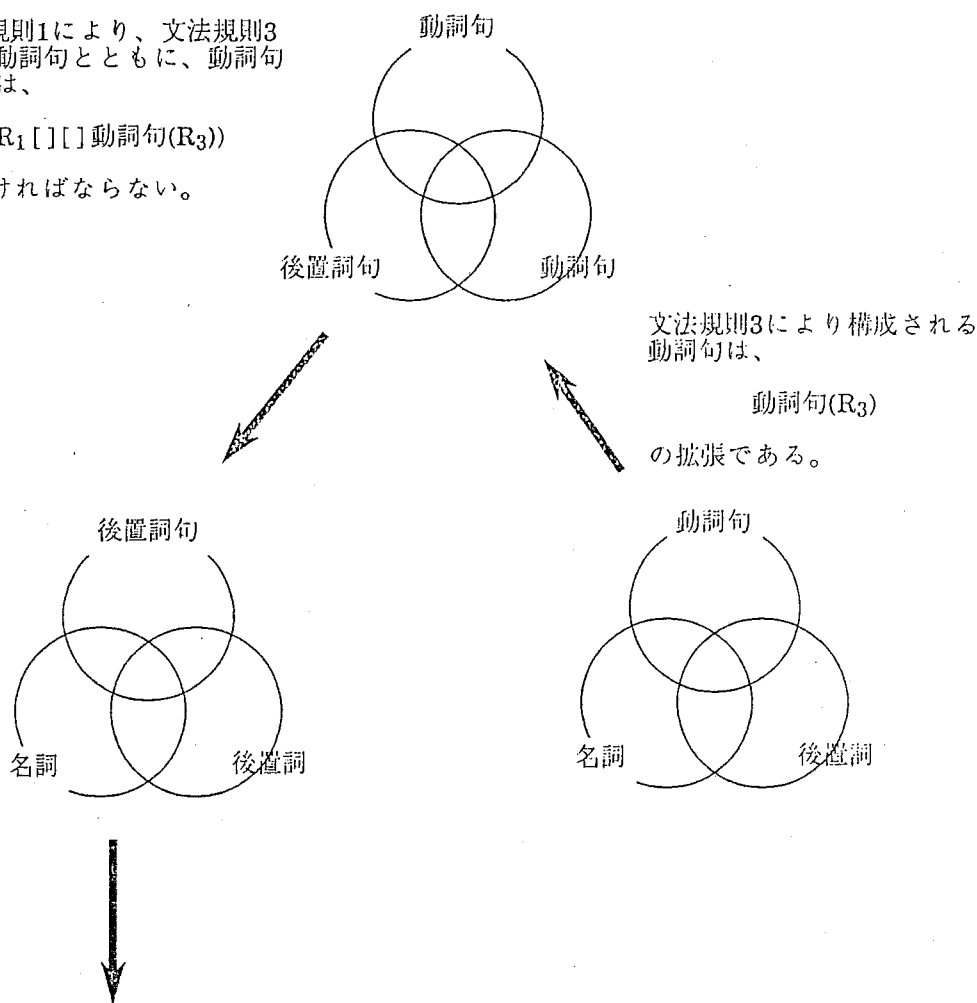
そして、こうした制約のもとで語彙記述を行うならば、その記述は、意図した句構造にそっての解析に関する限り(あるいは、さらにその結果意図した結果を得るという点に関する限り)妥当であるので、解析の試行により、この点に関しての検証を行う必要がないので、解析の試行に伴うはじめに述べたような困難を回避することができる。

ところで、[Fig.14]に示したように、ある素性構造 x と単一化可能であるという制約を遵守しての記述は、 x をグラフの形式に図式化し、この図式の上をなぞるように記述するならば、非常に簡単に、 x との単一化可能性を損なうことなく記述を進めることができる。なぜならば、その図式の上を擦っ

後置詞句が、文法規則1により、文法規則3により構成された動詞句とともに、動詞句を構成するためには、

後置詞句($\text{combine}(R_1 [] [] \text{動詞句}(R_3))$)

と単一化可能でなければならない。



L(情報)のみが確定しているものとして、他の語の語義が未知であるものとするれば、L(は)が句構造Pの終端記号となるためには、

後置詞($\text{combine}(R_2 \text{後置詞句}(\text{combine}(R_1 [] [] \text{動詞句}(R_3))) L(\text{情報}) []$)

と単一化可能でなければならない。

[Fig.11 L(情報)が確定した状態での、L(は)が句構造Pの終端記号であるための条件]

たり、新たなものを書き加える限りは、単一化可能性を損なうことはなく、この明確なガイドラインを超えなければ、単一化可能性を損なう記述をなし得ないからである。

8. 辞書記述の妥当性II

ここまでの議論では、各語彙項目の記述の妥当性として、それが特定の句構造中の特定の終端記号の値となるための妥当性のみを考えてきた。が、通常は、一つの語彙記述は、同じ一つの語彙項目に対応する異なる終端記号の値として併用される、ということ、無視する訳にはいかない。また、これまでは、ある句構造、あるいは、ある解析結果を得るための条件を考えてきたが、これとは逆に、ある句構造、あるいは、ある解析結果を排除する必要があることも考えねばならない。本章ではこれらの問題について簡単にふれておくことにする。

8.1 併用される語彙項目の満たすべき条件

いま、ある語彙項目の語彙記述 x が、異なる終端記号の値として併用されることを考えよう。これら記号各々の値の制約を、 l_0, l_1, \dots, l_n とすると、 x がこれらすべての値となるためには、 x は下式の

$$\text{unify}(l_0 l_1 \dots, l_n)$$

値の拡張でなければならない。逆にもしこの単一化が失敗するならば、一つの素性構造で、これら記号すべての制約を同時に満たすことはできない。

8.2 ある結果を排除するための条件

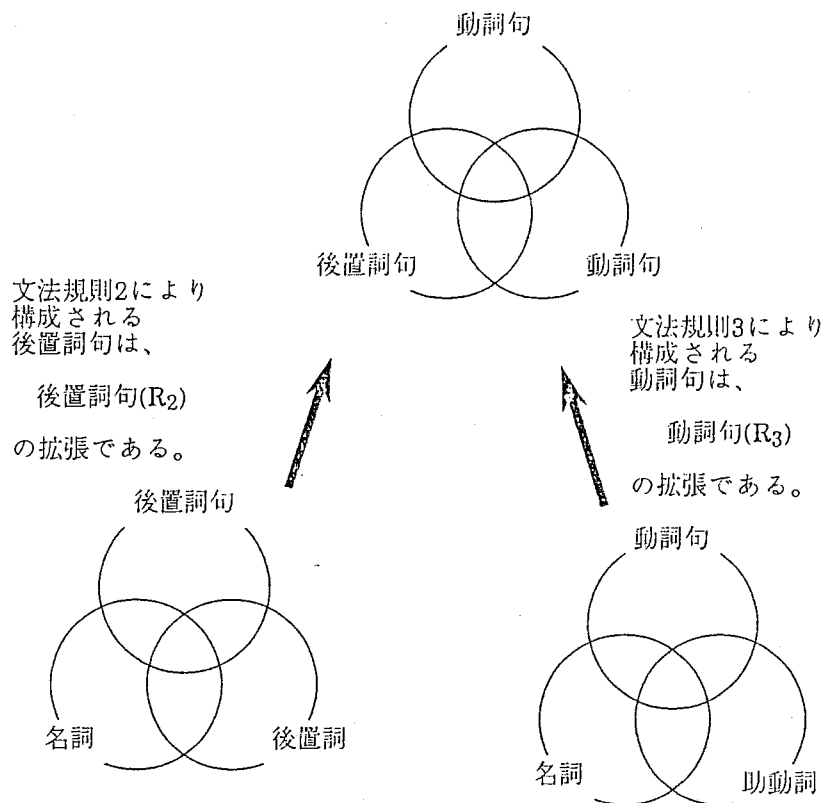
いま、ある文の句構造として、 P が望ましく、 Q は避けたいものとしよう。 P と Q は異なる構造であるので、各終端記号について、それが P を成すための条件と、 Q を成すための条件を計算すれば、

文法規則1に従って、後置詞句(R_2)の拡張と動詞句(R_3)の拡張によりつくられる動詞句の値 x は

$$\text{動詞句1}(\text{combine}(R_1 [] \text{後置詞句}(R_2) \text{動詞句}(R_3)))$$

の拡張である。ただしこのとき

$$\begin{aligned} & \text{動詞句1-後置詞句-動詞句2}(\text{combine}(R_1 [] \text{後置詞句}(R_2) \text{動詞句2}(R_3))) \\ = & \text{動詞句1-後置詞句-動詞句2}(\text{combine}(R_1 x [] [])) \end{aligned}$$



[Fig.12 句構造の開始記号の値の制約]

少なくとも、一箇所で両者が食い違うはずである。いま、ある単語に対応する記号で、Pを成すための条件をp、Qを成すための条件をqとして、

$$P \neq q$$

となったものとしよう。このとき、この単語の語義として、Pと単一化可能で、かつ、qと単一化できないものを選べば、この文の句構造として、Pを得、同時にQを排除することができる。

9. まとめ

辞書記述作業の中心となるのは、記述の妥当性を検証する作業である。通常、辞書記述の検証の手段として、検証の対象である記述を用いて、処理を試行することが行われている。ところが、記述の妥当性の検証の手段として、処理の試行には、

記述に問題がある場合の結果の解釈が容易ではない。

といった、問題点があり、これが辞書開発作業を困難なものにしている。

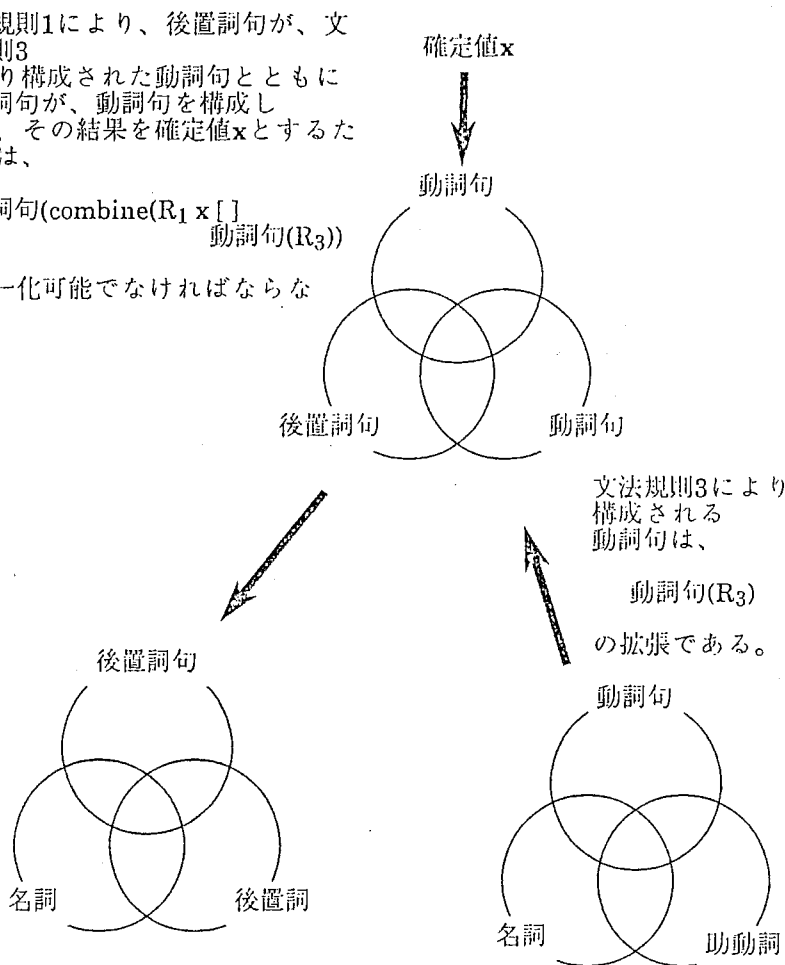
本稿では、少し視点を変えて、意図した結果を得るための個々の語彙項目に関する制約を明らかにし、その制約のもとで語彙記述を逐行することを提案した。そして、これを実現するための原理について述べた。こうして、妥当な制限のもとで辞書記述を進めることによって、記述の検証作業と、それに付随する上述の問題点を避けることができる。

今後は、本報告で述べたような戦略に基づく辞書開発環境を実現し、実際の運用を通じて、ここで述べた戦略の有効性を検証する必要がある。

文法規則1により、後置詞句が、文法規則3により構成された動詞句とともに後置詞句が、動詞句を構成し、かつ、その結果を確定値xとするためには、

後置詞句(combine(R_1 x [] 動詞句(R_3)))

と単一化可能でなければならない。

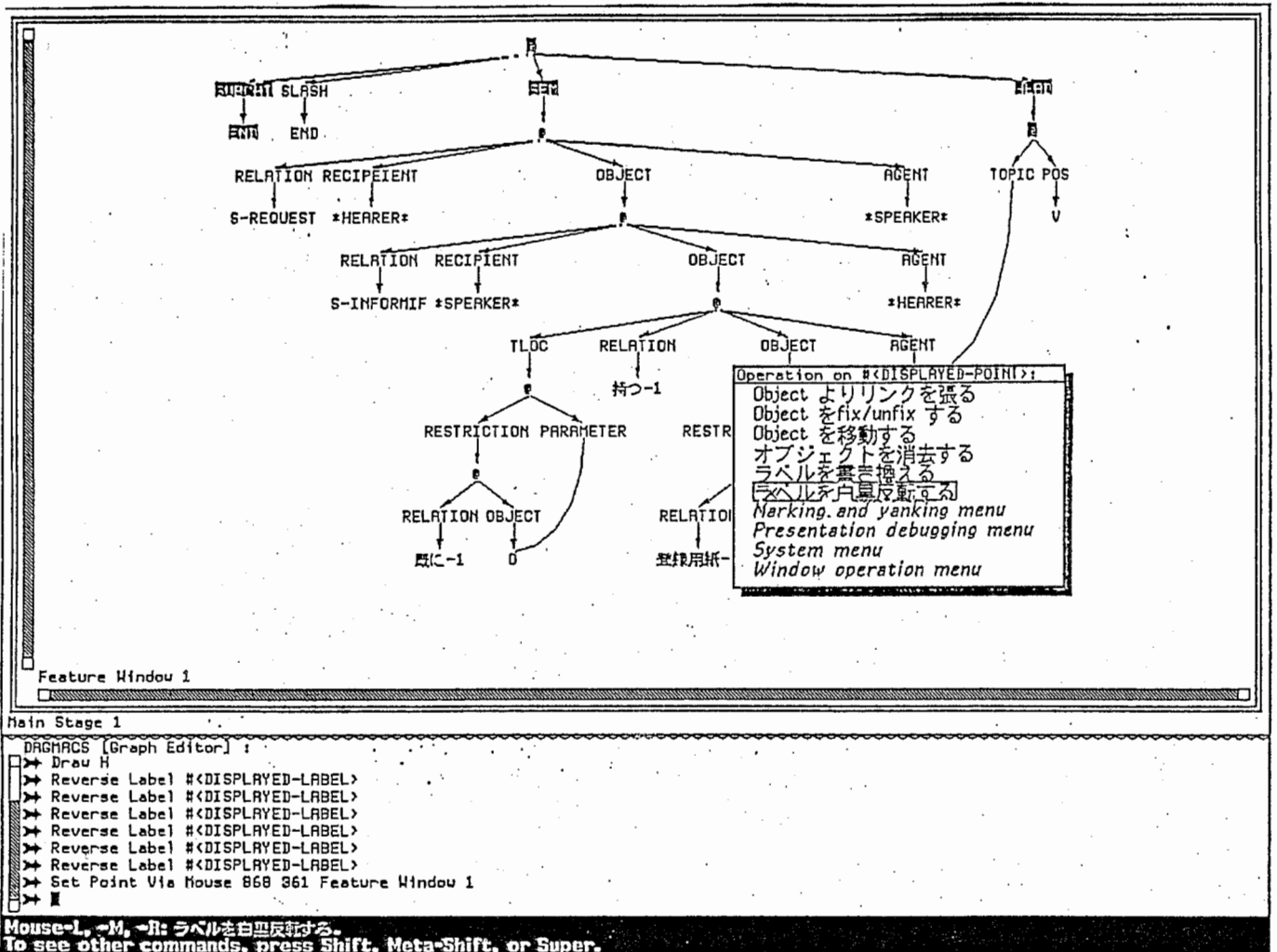


他の語の語義が未知であるものとするれば、L(情報)が、開始記号の値を確定値xとするこの句構造の終端記号となるためには、

名詞(combine(R_2 後置詞句(combine(R_1 x [] 動詞句(R_3)))) [] [] []

と単一化可能でなければならない。

[Fig.13 開始記号の値を確定値xとするために個々の終端記号に課せられる制約]



[Fig.14素性構造で表される制約のもとでの語彙記述の視覚化による支援]
 図式で表現された制約の上で、これをなぞるように語彙記述を遂行することで、制約をあらわす素性構造と単一化可能であるという条件のもとでの記述が極めて容易となる。