

制御情報をもつ選言記述形式とその単一化

新 納 浩 幸^{†*}

本論文では選言的素性構造の記述形式である補強 ENF とその上の単一化手法を提案する。単一化に基づく解析では、素性構造の選言をどのように記述し、またその記述形式上の単一化をどのように定義するかが効率を左右する。従来、Kasper は Kasper 標準形という記述形式を提案した。単一化処理は選言の展開を回避あるいは遅延することで効率を上げる。しかし選言が展開される場合、選言要素間の無矛盾性の検査が起こり、この処理の効率が非常に悪い。一方、Eisele は ENF という記述形式を提案した。この手法では単一化時に参照されない選言を展開しなくてよいため、上記の無矛盾性の検査がかなり減り、経験的に Kasper の手法よりも効率がよい。しかし ENF では Kasper 標準形のように、選言要素間の共通情報を明示できない。このため無駄な選言の展開が起こったり、選言の展開を回避するために行われる単一化に無駄な計算が生じる場合が多い。補強 ENF は、ENF に Kasper 標準形の長所を取り入れたものであり、ENF の選言表記部分にその選言の任意の共通情報を付加できる。単一化ではまずこの共通情報との無矛盾性を検査する。これによって、選言の無駄な展開が減るとともに、選言の展開を回避するために行われる単一化も効率化される。また本手法は ENF の特徴をそのまま継承するため、参照されない選言を展開することも回避できる。

Disjunctive Description with Control Information and Unification on It

HIROYUKI SHINNOU^{†*}

This paper presents a new description (Reinforced ENF) for disjunctive feature structures and a unification method on it. For efficient unification-based parsing, some important problems are how to represent feature structure containing disjunctive values and how to unify them. Kasper proposed general disjunctive form. Unification on this description can avoid or delay expansion to disjunctive normal form, but, it is expensive to check consistency among disjuncts. On the other hand, Eisele defined a description formula ENF. Unification on ENF doesn't expand non-referred disjunctive. So, experimentally Eisele's method is more efficient than Kasper's. However, Eisele's method has some wasteful expansions for disjunctive part, and has somewhat wasteful calculations in avoiding expansion. The Reinforced ENF is based on ENF and has merit of Kasper's method. This description can have any common information among disjuncts. The unification firstly checks consistency with common information. By the check, we can come over above ENF demerits.

1. はじめに

近年の文法記述形式として LFG, HPSG など単一化を基本としたものが数多く提案されている。これらの文法では語や句の構文意味構造を素性構造で表現し、文法自体も素性構造に変換し、解析処理では、文脈自由型のまとめ上げ規則と文法と語や句の単一化によって、文の構造を解析しようとする。基本的に上記

の方式はすべての曖昧性を導き、用意してある文法の条件を満たさない構造を順次削ってゆき、最終的に残ったものを文の解釈とする方式である。

しかし、この枠組では多数の意味を持つ語を扱ったり、キメの細かい解析結果を出そうとすると、ある部分列に対応する構文意味構造の曖昧性が爆発的に増えてしまい、現実的には処理が不可能になってしまう。

そこで、複数の素性構造を選言を用いることによって1つの構造（以下これを選言的素性構造と呼ぶ）として表し、それをそのまま単一化の処理にかけることができれば、解析効率が大幅に向上する。また、辞書、文法の記述容易性、記述量、人間による認可性も向上する。

このような背景をもとに選言的素性構造の記述形式

[†] 松下電器産業株式会社情報通信東京研究所
Tokyo Information and Communications Research
Laboratory, Matsushita Electric Industrial Co.,
Ltd.

* 現在 茨城大学工学部システム工学科
Presently with Department of Systems Engineering, Ibaraki University

とその上の単一化手法の研究が Kasper, Eisele らによって行われてきた^{1),2)}。

通常、単一化に基づく解析では単一化のほとんどが失敗する。これを選言的素性構造で考えると、選言的素性構造の選言要素となっている素性構造のほとんどが、単一化で失敗することに対応する。このため選言要素となっている素性構造をなんらかの情報で分類し、そのクラスの代表的情報によって、そのクラス全体に対する単一化の失敗を導くことが選言的素性構造の単一化には有効である。この点に重点をおいた手法が Kasper の手法である¹⁾。

Kasper は以下のような Kasper 標準形と呼ばれる記述形式を提案した。

$$\text{uconj} \wedge \text{disj}_1 \wedge \text{disj}_2 \wedge \dots \wedge \text{disj}_n$$

Kasper 標準形では選言的素性構造を確定部 (uconj) と不確定部とから構成する。確定部は選言の中で確定している情報を表し、不確定部は上記と同一の形式を持つ式を選言要素とする選言式 (disj_i) の連言として表す。Kasper 標準形上の単一化では確定部の情報を先に単一化することにより不確定部の持つ選言の展開を遅延あるいは回避できる。つまり Kasper 標準形は、上記の分類情報と代表的情報を選言要素間の共通情報である確定部に取ったものである。Kasper の手法では確定部との単一化の失敗によって矛盾が導かれる場合に効率が上がる。しかしその単一化が成功した場合、選言の無矛盾性のチェックをしなくてはならず、その処理で効率が悪くなる。このためにいくつかの提案がなされてきているが根本的な改善は難しい³⁾⁻⁵⁾。

一方、Eisele は ENF (Efficiency oriented Normal Form) という記述形式を提案した²⁾。これは曖昧部分のある属性の値に局所化する値選言 (value disjunct) という表記法である。これを実現するために、ENF では素性構造の相互参照 (coreference) の表現にパス表現と呼ばれる表記法を利用している。ENF 上の単一化は素性構造に対するグラフ単一化を拡張することで得られる。Eisele の手法の特徴は、単一化時に参照されない選言を展開しなくてよい点である。このため上記の無矛盾性のチェックがかなり減り、経験的に Kasper の手法よりも効率がよい^{*}。

しかし ENF では Kasper 標準形のように選言的素性構造に対してその構造の持つ代表的情報を明確な形で与えられない。このため無駄な選言の展開が起こったり、選言の展開を回避するために行われる単一化に無駄な計算が生じる場合が多い。

本論文では Eisele の手法をベースに Kasper の手法の長所を取り入れた選言的素性構造の記述形式とその上の単一化を提案する。基本的に ENF とその上の単一化を改良する。改良のポイントは、ENF の選言表記部分にさらにその選言の各要素が共通して持つ情報のいくつかを括り出すことである。選言部分の単一化の際に、まずその共通部分に対して単一化を行う。このときの単一化が失敗すると、その選言を展開しても単一化が失敗するために、選言の展開を行わずに単一化の失敗を察知することができる。

本論文の記述形式を用いると選言要素のある属性の値に注目して分類することができる。このとき注目している属性の値は上記の共通情報のように括り出している形になる。つまり選言要素間から括り出した共通情報は Kasper 標準形の確定部のように、選言要素を分類したクラスの代表的情報として明示することができる。このため本手法による単一化は、ENF 上の単一化と比べて無駄な選言の展開が減るとともに、選言の展開を回避するために行われる単一化も効率化される。また本手法は ENF の値選言の手法をそのまま用いているため、Kasper 標準形のように参照されない選言を展開する無駄も回避できる。

2. 選言記述形式 ENF とその問題点

2.1 選言記述形式 ENF

以下に Eisele の提案した選言的素性構造の記述形式 ENF を示す。ENF により記述される式を ENF 式と呼ぶ。αが ENF 式である (α∈ENF) ための必要十分条件は以下の2つの条件を満たすことである。(条件1) αは以下のいずれかのシンタクスである。

- (1) NIL
- (2) TOP
- (3) a where $a \in A$
- (4) $l_1 : \phi_1 \wedge \dots \wedge l_n : \phi_n$
where $\phi_i \in \text{ENF} - \{\text{TOP}\}$
 $l_i, l_j \in L,$
 $l_i \neq l_j$ for $i \neq j$
- (5) $\phi \vee \psi$ where $\phi, \psi \in \text{ENF}^*$
- (6) $\langle p \rangle$ where $p \in L^*$

* 選言的素性構造の単一化処理は、選言標準形の充足可能性の判定問題を内部に含むために NP 完全となり、どのように工夫した選言的記述方法を考案しても、最悪の場合には、計算量は選言要素の数の指数関数の大きさになる。このため理論的にどちらの手法が優れているかは示せない。

(条件2) α は ENF 条件を満たす.

条件1のシンタクスにおいて, A はアトムの集合, L はラベル(属性名)の集合である. 直観的に, 条件1のシンタクスは以下のように捉えられる.

- (1) NIL はすべての ENF 式と単一化可能なもの.
 $\alpha \wedge \text{NIL} = \alpha, \text{NIL} \wedge \alpha = \alpha$
 が成立する.
- (2) TOP はすべての ENF 式と単一化不可能なもの.
 $\alpha \wedge \text{TOP} = \text{TOP}, \alpha \wedge \text{TOP} = \text{TOP}$
 が成立する.
 単一化の失敗の結果値として利用できる.
- (3) a はアトム.
- (4) $l_i: \phi_1 \wedge \dots \wedge l_n: \phi_n$
 は属性名 l_i の値が ϕ_i であるような素性構造に対応する.
- (5) 選言の表記であり, ϕ あるいは ψ が成り立つことを示している.
- (6) $\langle p \rangle$ はパス表現 (path equivalence) である. 素性構造を有向グラフと考えて, p で示されるラベル(属性名)を辿った値と同じ値を持つことを表している.

条件2の ENF 条件は本論文の拡張部分と直接関係ないためここでは直感的な説明だけを行う. ENF 条件とは, ENF 式に対応する選言的素性構造を選言標準形に直した際, 各素性構造の持つパス表現のパスの各節点にアトムが与えられることがないという条件である. 例えば図1の素性構造はパス表現 $\langle b \ c \rangle$ の節点である属性 b の値にアトムが与えられているため, ENF 条件を満たしていない.

ENF 条件は ENF 上の単一化の正しさを保証するものであり, この条件にあまり神経質になる必要はない. 本来 ENF 条件を満たさない選言的素性構造は内部に矛盾を持っているため, 語や文法の素性構造の記述の際には通常その条件を破ることはない. また引数

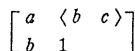


図1 ENF 条件を満たさない素性構造
 Fig. 1 A feature structure not to satisfy ENF condition.

* 本来の ENF では ϕ と ψ が TOP でないという条件があるが, これは ENF 上の単一化に何の変更も加えずに省くことができるため, 後の議論も考えて, ここではこの条件を付けない形で記述する.

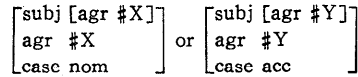


図2 ENF 式 α に対する素性構造
 Fig. 2 Feature structures corresponding to ENF formula α .

の ENF 式が ENF 条件を満たしていれば ENF 上の単一化によって得られる選言的素性構造も自動的に ENF 式を満たすので, 実用上は無視できる.

以下に ENF 式の例をあげる.

$$\alpha = \text{subj} : \text{arg} : \langle \text{agr} \rangle \wedge \text{case} : (\text{nom} \vee \text{acc})$$

上記の α は図2の2つの素性構造の選言を表している. 図2では相互参照の表記を通常のタグ付けによる表現で表している.

2.2 ENF 上の単一化

ここでは ENF 上の単一化を説明する.

ENF 上の単一化は, 通常の素性構造に対するグラフ単一化を拡張した形になっている. 基本的に引数である2つの ENF 式のラベルを比べ, 一方のみに現れるラベルに対しては, そのラベルとラベル値を単一化結果の ENF 式に付加し, 共通して現れるラベルに対しては, そのラベル値を再帰的に単一化し, その結果とラベルを単一化結果の ENF 式に付加する.

例えば, 式(1), (2)で示される ENF 式 α と β の単一化を考える.

$$\alpha = a : \phi \wedge b : \psi \tag{1}$$

$$\beta = a : \psi \wedge c : \phi \tag{2}$$

片方のみ現れるラベルとして b (α のみに現れる) と c (β のみに現れる) がある. それらのラベルとラベル値を単一化結果の ENF 式に付加する(式(3)).

$$b : \phi \wedge c : \phi \tag{3}$$

次に共通して現れるラベルには a がある. α と β の a のラベル値 ϕ と ψ を単一化して, その単一化結果(ここでは θ としておく) とラベル a を式(3)に付加し, α と β の単一化結果が得られる(式(4)).

$$a : \theta \wedge b : \phi \wedge c : \phi \tag{4}$$

次に選言の扱いを説明する. ENF では, 選言はあるラベルの値として現れる. 単一化の際に一方が選言であった場合, それを展開することで単一化が行われる. 例えば, 式(1)の α の a のラベル値 ϕ が, 式(5)の選言の形をしている場合, おのおのの ϕ_i が ψ と単一化される.

$$\phi = \phi_1 \vee \phi_2 \tag{5}$$

ϕ_i が ψ と単一化された結果を θ_i とすると, 式(6)で表される ENF 式 θ が ϕ と ψ の単一化結果となる.

$$\theta = \theta_1 \vee \theta_2 \quad (6)$$

次にパス表現の扱いを説明する。パス表現と ENF 式が単一化される場合、この ENF 式をパスで示される地点の条件として捉える。単一化結果はその地点を示すパス表現をひとまず返し、後でこの条件を検査する。

上記の処理を行うために ENF の単一化では、単一化の核になる関数 (unify_aux) に、局所的結果と大域的結果の 2 つを返値として持たせている。局所的結果は通常の単一化結果であり、大域的結果はパス表現によってもたらされた条件を溜めこんだものである。パス表現が出現しない場合、大域的結果は常に恒真 (NIL) である。unify_aux は大域的結果が恒真となるか偽 (TOP) となるまで求められた局所的結果と大域的結果を再帰的に単一化する。恒真になった場合、そのときの局所的結果が最終的な単一化結果となり、偽となった場合、その単一化は失敗 (TOP) となる。

上記に例として示した α と β の単一化は局所的結果だけに注目したものである。

また ENF 上の単一化の特徴はパス表現と選言の組み合わせの扱いにある。選言を展開して選言要素をおのおの単一化した際におのおのの大域的結果に違いがある場合、おのおのの局所的結果と大域的結果をその場で単一化し、それぞれの結果の選言を大域的結果として返す。この処理は本論文の主張する拡張部分と直接関係しないため、ここでの詳細な説明は省く。

2.3 ENF の問題点

選言の展開を回避、あるいは遅延させるという観点から単一化を効率化しようと考えた場合、以下の 2 点が重要である。

- (1) 選言要素となる複数の素性構造をある情報で分類すること。
- (2) 分類されたクラスのある代表的情報によってそのクラスに対する単一化の失敗を導くこと。

分類すること自体はどのような記述形式でも可能である。例えば、素性構造 A, B, C, D, E の選言を考え、ある情報によって A と B からなるクラス、C と D と E からなるクラスに分ける場合、括弧を用いて、

$$(A \vee B) \vee (C \vee D \vee E)$$

と表せばよい。つまり、分けられたクラスの代表的情報が即座に参照可能であることが本質的である。

選言的素性構造の記述形式を考える場合、クラスに属する素性構造を設定した記述形式でコンパクトに表

現でき、その表現中には明確な形で代表的情報が現れているのが理想である。ここで代表的情報というのは、通常、その選言に含まれる素性構造が持つ共通の情報である。

上記の考察を ENF にあてはめてみると、ENF ではそのクラス中の代表的情報を明確に与えるシンタクスを持っていないという問題がある。このため代表的情報への参照が困難である。またシンタクスを持っていないという以前の問題として、ENF では代表的情報になり得る共通情報でさえも陽に表すことができない場合が多い。それは ENF の選言をバックする能力が乏しく、選言要素間で曖昧部分が 1 つのラベルの下に集中しない限り、ENF では選言をフラットに並べた形でしか記述できないからである。

例えば A, B を図 3 のような素性構造とする。A と B は属性 b の属性 c の値が 3 という事で同一のクラスになっているとする。A, B に対する ENF 式は式 (7), (8) のようになる。

$$a : 1 \wedge b : c : 3 \wedge b : d : 4 \quad (7)$$

$$a : 2 \wedge b : c : 3 \wedge b : d : 5 \quad (8)$$

ここで式 (7), (8) の選言を ENF で表そうとすると、式 (9) のように選言をフラットに並べた形にしか記述できない。

$$(a : 1 \wedge b : c : 3 \wedge b : d : 4)$$

$$\vee (a : 2 \wedge b : c : 3 \wedge b : d : 5) \quad (9)$$

式 (9) ではクラスの共通式である $b : c : 3$ が陽に表されていない。つまり代表的情報が表されていない。

代表的情報が表されていないと、不要な選言の展開が生じる場合がある。例えば式 (9) と以下の式 (10) が単一化される場合を考える。

$$a : 1 \wedge b : c : 2 \wedge b : d : 4 \quad (10)$$

この場合、式 (9) は式 (7) と (8) に展開され、おのおのが式 (10) と単一化される。結果的におのおのの単一化は失敗し、全体としての単一化結果 (TOP) が得られる。一方、式 (9) において、 $b : c : 3$ が代表的情報であると明示されていれば、まずその式と式 (10) との無矛盾性をチェックすることで、選言を展開せずに、この場合の単一化結果が得られる。

本質的には ENF では代表的情報を明確に与えるシンタクスを持っていないことが問題である。

$$A = \begin{bmatrix} a & 1 \\ b & \begin{bmatrix} c & 3 \\ d & 4 \end{bmatrix} \end{bmatrix} \quad B = \begin{bmatrix} a & 2 \\ b & \begin{bmatrix} c & 3 \\ d & 5 \end{bmatrix} \end{bmatrix}$$

図 3 素性構造 A と B

Fig. 3 Feature structure A and B.

$$B' = \begin{bmatrix} a & 2 \\ b & [c & 3] \\ & & d & 4 \end{bmatrix}$$

図 4 素性構造 B'

Fig. 4 Feature structure B'.

例えば図 3 の B の一部を変更して図 4 に示す B' の素性構造と A との選言を考える。B' に対応する ENF 式は以下の式 (11) である。

$$a : 2 \wedge b : c : 3 \wedge b : d : 4 \quad (11)$$

この場合、式 (7) と (11) の選言は属性 a の下に局所化できるため、式 (12) の ENF 式で表記できる。

$$a : (1 \vee 2) \wedge b : c : 3 \wedge b : d : 4 \quad (12)$$

このとき $b : c : 3$ と $b : d : 4$ が A と B' の共通情報を担っていることはわかるが、 $b : c : 3$ が代表的情報であると明示することができない。

代表的情報が特定できないと、無矛盾性のチェックに不必要な計算が起こる場合がある。例えば、式 (12) と式 (10) の単一化を考える。まず、式 (12) の代表的情報に対応する式 (13) が式 (10) と単一化される。

$$b : c : 3 \wedge b : d : 4 \quad (13)$$

この単一化は失敗し、選言の展開は避けられるが、もしも $b : d : 4$ と式 (10) が先に単一化されていた場合、この部分の計算は無駄である。グラフ単一化では素性構造の奥にある情報ほど遅れて単一化される。このため、もしも代表的情報が奥深くに存在している場合、無駄な計算を行う可能性が高い。

一方、Kasper 標準形では確定部のとり方が共通情報であれば十分であるために、式 (12) は式 (14) の形で記述できる。

$$b : c : 3 \wedge (b : d : 4 \wedge (a : 1 \vee a : 2)) \quad (14)$$

式 (14) では $b : c : 3$ が代表的情報として明示されている。

3. 補強 ENF とその上の単一化

3.1 補強 ENF

以下に本論文の提案する選言的素性構造の記述形式である補強 ENF を示す。補強 ENF により記述される式を補強 ENF 式と呼び、 α が補強 ENF 式である ($\alpha \in$ 補強 ENF) ための必要十分条件は以下の 2 つの条件を満たすことである。

(条件 1') α は以下のいずれかのシンタクスである。

- (1) NIL
- (2) TOP
- (3) a where $a \in A$
- (4) $l_1 : \phi_1 \wedge \dots \wedge l_n : \phi_n$

where $\phi_i \in$ 補強 ENF - {TOP}

$l_i, l_j \in L,$

$l_i \neq l_j$ for $i \neq j$

(5) $[\theta; \phi \vee \psi]$

where $\phi, \psi \in$ 補強 ENF

θ は単純式

θ は ϕ と ψ の共通情報条件を満たす。

(6) $\langle p \rangle$ where $p \in L^*$

(条件 2') α は補強 ENF 条件を満たす。

上記で用いられた単純式、共通情報条件、補強 ENF 条件の定義を行う前に、条件 1 の (5) 式を補強選言式と呼び、(5) 式の θ の部分を補強式と呼ぶことにしておく。

「 θ が単純式である」とは θ が条件 1' の (1), (3), (4) 式から構成され得る場合をいう。つまり式の中に選言とパス表現が含まれない場合をいう。また「 θ は ϕ と ψ の共通情報条件を満たす」とは、 ϕ の選言標準形の各要素と ψ の選言標準形の各要素が θ を包摂 (subsume)⁶⁾ することである。直感的には θ が ϕ と ψ の間で確定している情報、共通に持っている情報と捉えられる。また厳密にはここで補強 ENF 式が選言標準形に展開可能であることを示す必要があるが、補強 ENF 式中の選言部分から補強式を取り除くと ENF 式に書き換え得ることから明らかであるため省略する。次に補強 ENF 条件であるが、これは今述べたように補強 ENF 式が ENF 式に書き換え可能であり、書き換えた場合のその ENF 式が ENF 条件を満たすことである。

補強 ENF 式の例として式 (9) の ENF 式を補強 ENF 式で表すと例えば式 (15) のようになる。

$$[b : c : 3; (a : 1 \wedge b : c : 3 \wedge b : d : 4) \vee (a : 2 \wedge b : c : 3 \wedge b : d : 5)] \quad (15)$$

Kasper 標準形と異なるのは補強式の右にある選言の式の中に $b : c : 3$ という情報が残されている点である。

また、補強 ENF 式は補強式である θ として ϕ と ψ の包摂関係における最大の共通情報である必要はない。このため ϕ と ψ の共通情報のうち解析で有用となるものだけを補強式に付加すればよく、まったく共通情報をとりださない、あるいは取り出せない場合は、補強式を NIL とすればよい。

上記の性質を利用すれば、式 (12) は式 (16) のような補強 ENF 式にも変形可能である。

$$[b : c : 3; (a : (1 \vee 2) \wedge b : c : 3 \wedge b : d : 4) \vee \text{TOP}] \quad (16)$$

これは ENF 上では行えなかった代表的情報の明示が達成されている。

3.2 補強 ENF 上の単一化

補強 ENF 上の単一化は補強選言式の補強式を無視することで ENF 上の単一化とまったく同じ処理で行える。本論文が提案する補強 ENF 上の単一化も ENF 上の単一化と基本的に同じである。ただし、単一化の一方の式が補強選言式である場合に補強式を補助的に用いて単一化の効率化を図る。

補強選言式と補強 ENF 式の単一化の際に、まず補強式と補強 ENF 式を単一化する。この単一化が失敗した場合には選言を展開せずに単一化結果である TOP が得られる。また成功した場合には、補強選言式の選言要素であるおのおのの補強 ENF 式と単一化相手であった補強 ENF 式を単一化し、おのおのの結果から ENF の単一化処理と同様の処理を行う。

例として式 (15) と式 (10) の単一化を考える。まず補強式 $b : c : 3$ と単一化相手の式 (10) を単一化する。この単一化は失敗するために選言の展開を行う前に式 (15) と式 (10) の単一化結果 TOP が得られる。

また補強 ENF は補強式で代表的情報を明示できるため、代表的情報との無矛盾性の検査に無駄が少ない。例えば式 (16) と式 (10) の単一化を考えた場合、ENF で起こり得た $b : c : 4$ と式 (10) との無駄な単一化が回避できている。

一方、補強 ENF 式は式の中の任意の式を括り出すことができるために、解析中に無用に式が大きくなってゆくこともある。このため、以下の変形式を適時行うことにする。

- $[NIL; [\alpha; \phi \vee \phi] \vee [\alpha; \theta \vee \lambda]] \Rightarrow$
 $[\alpha; [NIL; \phi \vee \phi] \vee [NIL; \theta \vee \lambda]]$
- $[\alpha; \phi \vee \text{TOP}] \Rightarrow \phi$

また補強選言式との単一化が成功した場合には、展開された選言要素の中に再び先の補強選言式が含まれているために、二重の計算を行うこともある。しかし補強式は単純式に限定しているため、補強式との単一化は計算コストが低く、二重になる計算のコストは、本手法により削減された計算処理のコストに比べ少ないと予測している。

4. 実験

本手法の有効性を計るために小規模な実験を行

った。

まず 40 個強の書き換え規則を持つ単一化文法と利用する語からなる最低限の辞書を作成し、それらを ENF に変換して ENF 上の単一化によって解析した。次に先の文法や語を分析し、曖昧性解消に有益であるような情報を分類情報として補強 ENF に書き換え、補強 ENF 上の単一化によって解析した。

解析例文は以下の 4 つの例文を用い、パーサは CKY パーサ⁷⁾を用いた。実験は、SPARC station 1 の KCL 上で、下記に示した例文をおのおの 10 回解析し平均時間を測定した。用いた辞書の各単語は平均 2 つの選言を持っている。また文法も各書き換え規則に対して平均 2 つの選言を持っている。

例文

- (a) My father wants to play tennis with me
 - (b) He was a pilot when he was young
 - (c) I saw a man with a telescope on the park
- 実験の結果は表 1 に示すとおりであった。

この実験では、補強 ENF を用いた単一化の方が ENF を用いた単一化よりも 2 倍弱高速であった。

5. 考察

本手法が従来手法 (ENF とその単一化処理) よりも効率化されるかどうかは、選言の中からのどの程度有効な補強式を選択できるかに依存するが、この処理を解析中にダイナミックに行うことは困難である。これは素性構造間の共通情報を見つけるのが困難であるだけでなく、より有効な表記を行う場合、曖昧性解消にどの情報が有効に利用できるかを予測することも必要となるからである。

本来、ENF の表記法とその単一化は、この困難性を避けている点も特長として持つ。実験でもこの点は踏襲し、選言がフラットに並んだ形からは、特に補強式を取り出していない。ただし、補強式との単一化の検査を通過して得られた選言では、その補強式は再び選言の各要素の共通情報となっているため補強式として選択することができる。

表 1 従来手法との比較
Table 1 Comparison with traditional method.

	従来手法	本手法
例文 (A)	10.850	5.767
例文 (B)	5.117	3.017
例文 (C)	11.800	7.400

実行時間 (秒)

```

if (Attri == a) then [処理A]
                else [処理B]
                ↓ ↑
[NIL; [Attri: a; [処理A]∨TOP]
      ∨ [処理B]]

```

図5 手続きと補強 ENF との関係
Fig. 5 Relation to Reinforced ENF and procedure.

```

[NIL;
 [Attri: a; [Aグループ辞書情報]∨TOP]
      ∨
 [Attri: b; [Bグループ辞書情報]∨TOP]
]

```

図6 補強 ENF 式による辞書記述
Fig. 6 Description of dictionary by Reinforced ENF.

以上の議論は解析中に行われる動的な処理であるが、ある単語の辞書情報に対応する素性構造や文法に対応する素性構造は固定されているため、あらかじめ、より効率的な補強 ENF 形式に変換しておくことができる。

例えば、 $\langle S \rangle \longleftrightarrow \langle Np \rangle \langle Vp \rangle$ という書き換え規則の持つ素性構造が、いくつかの選言で表される場合を考えてみる。文法の書き換え規則が起こった場合の処理を考えると、「まず $\langle Vp \rangle$ のテンスで場合分けして、次に $\langle Np \rangle$ が代名詞かどうかで場合分けして、…」といった処理になる場合が多い。これはプログラムの場合には、if-then 文が入れ子構造になっている場合に相当する。文法の処理上の意味がわかっているならば、if-then 文の条件や、if-then 文の入れ子構造の順序を考慮することで、解析の効率が上がるような記述が可能である。

補強式を選択することは、この if-then 文の入れ子構造を作ることに対応する(図5)。つまり補強 ENF 式は、if-then 文の入れ子構造のマクロと見ることができ、文法の処理上の意味を考えて、効率的な記述を行うことが可能になる。また辞書情報に対する素性構造も曖昧性解消に有効であるような属性名に注目して選言をグループ分けしておくこともできる(図6)。

6. 関連研究

単一化文法によって解析を行う場合、その文法に選言や否定に対応するものが導入されると極端に効率が悪くなる。文法を規則的に書き直せば改善されるが、それでは単一化文法の持つ宣言的記述の長所がなくなってしまう。そこで宣言的な記述を保持したまま効率を上げるために、文法に制御情報を付加してゆくこと

が提案されている⁸⁾。本論文で提案している記述形式も補強式という形で単一化文法に制御情報を付加したものと見ることができる。

近年、素性構造に型を導入した型付素性構造が提案されている⁹⁾。これは素性構造をある軸によって分類し、その分類したクラスに名前(型)をつけたものである。単一化の際には型同士の無矛盾性から検査されるので、単一化の失敗が早期に発見されることになり効率的である。この型も一種の制御情報と見なすことができ、本論文でも、型付素性構造で行われた分類の軸を補強式に取ることと同様の効果が得られる。しかも補強式は任意の軸による分類が可能になっているために、より柔軟な制御が可能になる。

選言的素性構造の単一化の効率化に関しては、パス表現と選言との単一化の際に生じる大域的な条件の影響範囲に注目する手法もある^{10)~12)}。ここではこの点に関する効率化については述べなかったが、本論文の手法はそれらの手法と相反するものではなく、統合できるものである。一般に語句や文法の構造をどのように設計するかで効率化の重点も変わってくる。ここでは相互参照の構造(パス表現)をあまり持たないような木構造に近い素性構造を念頭に置いた。実際4章の実験では基本的にパス表現が、あまり単一化時に現れないように語句構造や文法を設計している。パス表現を全く持たない素性構造に対する選言的素性構造の記述形式に関しては、本論文とは異なる Kasper 標準形と ENF を統合したものも提案され¹³⁾、かなりの効率化がなされている。しかし単一化文法を記述する場合にはパス表現を完全に排除することは難しく、パス表現が記述できる形式も必要である。

またここでの単一化はグラフ単一化を基本としているが、項単一化の特徴を生かした選言的素性構造の単一化の効率化の研究もある¹⁴⁾。

7. おわりに

本論文では選言的素性構造の記述形式である補強 ENF とその上の単一化手法を提案した。

補強 ENF は従来の代表的記述形式である ENF を改良したものである。補強 ENF では、選言要素をある代表的情報によって分類し、しかもその代表的情報を明示できる。このため ENF 上の単一化と比べて無駄な選言の展開が減るとともに、選言の展開を回避するために行われる単一化も効率化される。これは従来のもう一つの代表的手法である Kasper 標準形の持つ

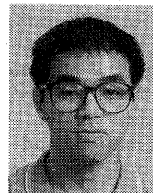
長所を加えた形になっており、ENF 上の単一化よりも効率化される。

本論文の手法はパス表現をあまり含まない素性構造に対する単一化に有効に働く。パス表現が頻出するような素性構造を扱う場合は、パス表現からの影響の局所性を調べる手法と融合すべきである。

参 考 文 献

- 1) Kasper, R.: A Unification Method for Disjunctive Feature Descriptions, *Proc. Annual Meeting of Association for Computational Linguistics-87*, pp. 235-242 (1987).
- 2) Eisele, A. and Dörre, J.: Unification of Disjunctive Feature Descriptions, *Proc. Annual Meeting of Association for Computational Linguistics-88*, pp. 286-294 (1988).
- 3) Carter, D.: Efficient Disjunctive Unification for Bottom-Up Parsing, *COLING-90*, Vol. 3, pp. 70-75 (1990).
- 4) 永田昌明, 小暮 潔: 文法規則の構造共有による選言的素性構造単一化の効率化, 情報処理学会第40回全国大会論文集, 2E-5 (1990).
- 5) 中山圭介, 松本裕治, 長尾 真: 選言的素性構造を用いた自然言語処理の曖昧性の表現とその解消方法, 第43回情報処理学会全国大会論文集, 3G-2 (1991).
- 6) 今村 誠: 素性構造の単一化, 情報処理, Vol. 32, No. 10, pp. 1070-1078 (1991).
- 7) 野村浩郷: 自然言語処理の基礎技術, 電子情報通信学会編, コロナ社, 東京 (1988).
- 8) Uszkoreit, H.: Strategies for Adding Control Information to Declarative Grammars, *Proc. Annual Meeting of Association for Computational Linguistics-91*, pp. 237-245 (1991).
- 9) Emele, M. C. and Zajac, R.: Typed Unification Grammars, *COLING-90*, Vol. 3, pp. 293-298 (1990).
- 10) Maxwell, J. T. and Kaplan, R.: An Overview of Disjunctive Constraint Satisfaction, *Proceedings of the International Workshop on Parsing Technologies*, pp. 18-27 (1989).
- 11) Dörre, J. and Eisele, A.: Feature Logic Disjunctive Unification, *COLING-90*, Vol. 2, pp. 100-105 (1990).
- 12) Strömbäck, L.: Unifying Disjunctive Feature Structures, *COLING-92*, pp. 1167-1171 (1992).
- 13) 菅野祐司, 長尾健司: 曖昧さの効率的処理のための構文解析手法について, 情報処理学会自然言語処理研究会報告, NL 74-6, pp. 41-48 (1989).
- 14) Nakano, M.: Constraint Projection: An Efficient Treatment of Disjunctive Feature Descriptions, *Proc. Annual Meeting of Association for Computational Linguistics-91*, pp. 307-314 (1991).

(平成4年9月4日受付)
(平成5年9月8日採録)



新納 浩幸 (正会員)

1961年生。1985年東京工業大学理学部情報科学科卒業。1987年同大学院理工学研究科情報科学専攻修士課程修了。同年富士ゼロックス、翌年松下電器を経て、1993年4月より茨城大学工学部システム工学科助手。自然言語処理の研究に従事。人工知能学会、ソフトウェア学会各会員。