

## MSLR 法を考慮した音声認識用日本語文法 -LR 表工学 (3)-

田中穂積 竹澤寿幸 衛藤純司

東京工業大学大学院情報理工学研究所 ATR 音声翻訳通信研究所 ランゲージウェア

〒152 東京都目黒区大岡山 2-12-1 〒619-02 京都府相楽郡精華町光台 2-2 〒115 東京都渋谷区上原 2-29-7-303

E-mail: tanaka@cs.titech.ac.jp takezawa@itl.atr.co.jp KYM04074@niftyserve.or.jp

筆者らは、一般化 LR 法で用いる LR 表を修正して自然言語の統語過程を柔軟に制御するさまざまな手法を開発している。その中に、隣接する前終端記号間の接続に関する制約を LR 表に組み込む手法がある。これは、CFG 規則と接続表を個別に記述し、両方の制約を LR 表に組み込む方法である。これによって、CFG 規則の一般性を保ちながら精密な自然言語解析が可能になる。この手法を音声認識用日本語文法に応用し、CFG 規則数を大幅に減らし、文法が見通しよく記述可能となり、保守や文法体系の拡張が容易になることを示す。

GLR 法、CFG 規則、接続表、LR 表、音声認識

### Japanese Grammar for Speech Recognition Considering the MSLR Method

Hozumi TANAKA† Toshiyuki TAKEZAWA‡ Junji ETOH††

†Graduate School of Information Science and Engineering, Tokyo Institute of Technology  
2-12-2 Ookayama, Meguro-ku, Tokyo 152, Japan

‡ATR Interpreting Telecommunications Research Laboratories  
2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02, Japan

††Languageaware 2-29-7-303 Uehara, Shibuya-ku, Tokyo 115, Japan

E-mail: tanaka@cs.titech.ac.jp takezawa@itl.atr.co.jp KYM04074@niftyserve.or.jp

This paper describes a new method of generalized LR parsing and its application to building a Japanese grammar for speech recognition. In this method, the phrase structure rules and the connections between neighboring morphemes are maintained separately and integrated into an LR table. It enables the flexible and fine-grained analysis of natural language. We applied this method to building a Japanese grammar for speech recognition and have achieved both drastic reduction of the number of the rules and improvement of the efficiency of the grammar.

GLR method, CFG rules, connection table, LR table, speech recognition.

#### 1 はじめに

自然言語解析の統語解析用のアルゴリズムとして一般化 LR (GLR) 法 [Tomita86] がある。Knuth による LR 法 [Knuth65] を文脈自由文法 (CFG) を扱えるように一般化したもので、最近では音声認識にも利用されるようになってきた。GLR 法では、統語解析に先立ち、与えられた CFG から解析用の表 (LR 表) をあらかじめ抽出しておき、この LR 表を用いて統語解析を進めていく。GLR 法は経験的に最も効率の良いアルゴリ

ズムであるとされている。

筆者らは、GLR 法で用いる LR 表を修正して自然言語の統語過程を柔軟に制御するさまざまな手法 (これらを総称して「LR 表工学」と呼ぶ) を開発している。本稿では、その中で、隣接する (前) 終端記号間の接続に関する制約を LR 表に組み込む新しい方法を説明し、この方法をベースにした音声認識用日本語文法について報告する。

- (0)  $S \rightarrow Z$
- (1)  $V \rightarrow \gamma v_i$
- (2)  $V \rightarrow \delta v_j$
- (3)  $W \rightarrow v_i \zeta$
- (4)  $W \rightarrow v_j \eta$
- (5)  $X \rightarrow \alpha V$
- (6)  $Y \rightarrow W \beta$
- (7)  $Z \rightarrow X Y$

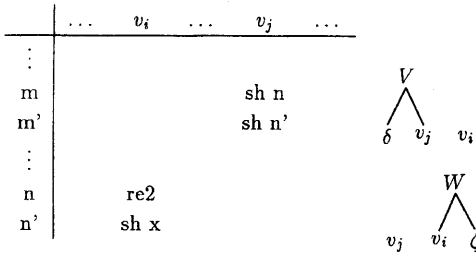


図 1: CFG と LR 表

## 2 MSLR 法の概略

自然言語の統語解析で用いる文法的な枠組みとして、これまで CFG がよく使われてきた。一方、形態素解析では、隣接する形態素間の接続可能性という局所的な制約が使われてきた。この局所的な制約を、以下では接続表とよぶことにする。この形態素間の接続制約を、CFG を使って記述できないわけではないが、いくつかの問題がある。たとえば、接続制約を CFG によって精密に記述しようとするとき多数の新しい非終端記号を導入しなければならないとなり、それにより CFG 規則の数が組合せ的に増大する [田中 95]。

この問題を解決するために、LR 表工学では、CFG 規則と接続表を個別に記述し、両方の制約を LR 表に組み込む。後述するように、CFG 規則から LR 表を作成する際に、接続表で許されないアクションを LR 表からすべて削除するのである。それにより、以下の二つが可能になる。

- (1) CFG 規則の記述者は、接続可能性を考慮した新たな非終端記号の導入を行なうことなしに、CFG 規則を記述することができる。
- (2) 接続可能性を表す接続表の記述者は、CFG 規則とは独立に接続表を記述することができる。

このようにして作成された LR 表は、これまで異なるレベルの制約であるとみなされてきた形態素間の接続制約と統語的な制約をともに含んでいるので、形態素解析と統語解析を GLR 法の枠組で統一的に扱うことが可能となる。この手法を以下では MSLR 法 (Morphological & Syntactic LR 法) と呼ぶ [植木 95]。

## 2.1 接続表の制約の LR 表への組み込み

終端記号の集合  $V_t = \{v_1, v_2, \dots, v_n\}$  の各要素を先読み語とする。そして、 $v_i$  と  $v_j$  がこの順に接続不可能で、 $v_i v_i, v_i v_j, v_j v_j$  という接続が可能であるとする。

$$\begin{aligned}
 \text{connect}[v_i, v_i] &= 1, & \text{connect}[v_i, v_j] &= 1 \\
 \text{connect}[v_j, v_i] &= 0, & \text{connect}[v_j, v_j] &= 1
 \end{aligned}$$

図 1 の (1) から (7) に与えた CFG から、図 1 の下部に示す LR 表が得られたと仮定する。

この LR 表に接続表の制約を以下のようにして組み込むことができる。状態 m で  $v_j$  をシフトするアクション sh n を実行して到達した状態 n において、先読み語が  $v_i$  の時に行なうアクション re2 は、 $V \rightarrow \delta v_j$  という規則の適用を意味している。この場合にはシフトした語  $v_j$  と先読み語  $v_i$  とがこの順に接続することになるが、接続表からこの接続は許されていないので、re2 (状態 n で先読み語が  $v_i$  のときに実行するアクション) を LR 表から削除することができる。

同様に、たとえば、 $W \rightarrow v_i \zeta$  を将来適用するために、状態 m' で sh n' を実行し、 $v_j$  をシフトして到達した状態 n' で、次の先読み語が  $v_i$  の時に実行するアクション sh x も LR 表から削除することができる。

以上に述べた二つのアクションの削除の例では、いずれも  $v_j$  をシフトした直後に実行するアクション (レデュースとシフト) を削除している。そこで、接続表の制約に従って LR 表からアクションを削除する手続きを一般化して次のように述べるができる。

アクション Act の直前に実行するアクションがシフトアクション Sh であり、Sh の先読み語が a、Act の先読み語が b であるとき、 $\text{connect}[a, b] = 0$  なら Act を削除する。

上記した手続きにより、接続表で許されないアクションを LR 表からすべて削除し、接続表を表す接続可能性に関する制約を LR 表に組み込むことができる。

### 2.1.1 制約伝播によるアクションの削除

前節に述べた手続きにより、接続表で許されないアクションを削除した結果、LR 表の他のアクションの削除が可能になることがある。これを制約伝播とよぶ。

たとえば、2.1 の最後に述べたアクションの削除により、LR 表上で、アクションがすべて削除された状態 (このような状態を空状態とよぶ) ができてしまうことがある。空状態にシフトするアクションがあるとき、空状態へシフトしてもその後の解析が続けられないので、空状態へシフトするシフトアクションを削除することができる。

これを一般化して述べると次のようになる。Act とよばれるアクションを実行した直後に実行すべきアクショ

ンが一つもなければ、Actの実行直後にエラーが生じるのは明らかである。したがって、このActを実行しても解析が続けられないのであるから、これを削除してよい。一方、Actを実行する直前に実行すべきアクションが一つもなければ、このActは永久に実行されない無効なアクションであるから、このアクションを削除することができる。

以上の手続きをまとめると次のようになる。

- (a) LR表の初期化手続き：LR表中のすべてのアクションに対して、以下を行なう：アクションActの直前に実行するアクションがシフトアクションShであり、Shの先読み語がa、Actの先読み語がbであるとき、 $connect[a, b] = 0$ なら、Actを削除する。
- (b) 制約伝播手続き：以下の手続きを、削除すべきアクションがなくなるまで繰り返す。

アクションActについて、Actの直前または直後に実行するアクションが一つもなければ、そのActを削除する。

手続き(a)により、接続表の制約を満たさないアクションが削除され、手続き(b)により新たなアクションが削除される。手続き(b)によりさらに次のアクションの削除が促されることがある。こうして、手続き(b)を繰り返す過程で、アクションの削除が連鎖的に伝播する。これを制約伝播とよび、制約伝播により接続表のもつ制約を完全にLR表に組み込むことが可能になる。

上記した方法は、従来のLR表作成アルゴリズムを用いてLR表を作成してから、接続表の制約を組み込み、制約伝播を施すものであった。この方法によれば、文法規則の数が増えるにつれて、はじめに作成するLR表の状態数が膨大になるため、効率の良い方法であるとはいえない。われわれは、LR表作成アルゴリズムを変更し、はじめのLR表を作成する段階で、接続表の制約を組み込む方法を開発し使用している。その詳細は文献[田中95]を参照されたい。

### 3 音声認識用日本語文法への応用

音声認識と言語解析を統合しようとする試みがさまざまに行われているが、両方で用いられる文法の、使用目的の違いに由来する粒度や体系の差異がその妨げになっている。

音声認識では、次に来る音素を予測することにより探索空間を狭めて音素認識の精度をあげることが目指される。文法もまたそのために用いられるので、音素や形態素のあいだの接続可能性に関する制約をできるだけ精密に記述することに最大の労力が払われる。その結果、品詞が細分化され、文法体系が複雑なものになる傾向がある。

これに対して、言語解析においては、文法は入力文に対して統語的・意味的に適切な構造を与えるために用いられるので、言語学的な自然さや、ルール的一般性・普遍性を実現することが重視される。その結果、品詞の過度の細分化は避けられ、文法体系は単純なものになる傾向がある。

音声認識と言語解析を統合するためには、文法体系のこのような矛盾は小さければ小さいほどよい。しかし、多くの音声認識用文法がそうであるように、形態素間の接続可能性というきわめて局所的な制約をCFGで記述しようとするかぎり、この矛盾は拡大することはあっても縮小することはない。MSLR法は、音声認識と言語解析を統合するという試みが当面するこのような課題に対してもごく自然な解決を与えることができる。

以下では、ATR音声翻訳通信研究所で開発された音声認識用日本語文法[竹澤95]を例に、MSLR法による解決法を説明する。一つは数詞の規則であり、もう一つは文末述語の規則である。いずれも、MSLR法の効果が最もよく発揮される言語表現である。

#### 3.1 数詞の規則

日本語の数詞の読みは、数字と桁単位との音声的な制約が錯綜していて、規則化することはかなり面倒な作業になる。たとえば、百の桁単位は「ヒャク」「ビャク」「ビャク」という三つの読みがあって、200は「ニヒャク」、300は「サンビャク」、600は「ロツビャク」というように、前に来る数字によって読みが定まっている。あるいは、1000は「セン」とも「イッセン」とも言うが、100は「ヒャク」としか言わない。

このような数詞の読みをCFGの枠組みで規則化しようとする、次のように、おびただしい数の規則が必要になる。

ichi	→ イチ	zyuu	→ ジュウ
iq	→ イッ	zyuq	→ ジュッ
ni	→ ニ	hyaku	→ ヒャク
san	→ サン	hyaq	→ ヒャッ
shi	→ シ	byaku	→ ビャク
yon	→ ヨン	byaq	→ ビャッ
go	→ ゴ	pyaku	→ ピャク
roku	→ ロク	pyaq	→ ピャッ
roq	→ ロッ	sen	→ セン
shichi	→ シチ	zen	→ ゼン
nana	→ ナナ	man	→ マン
hachi	→ ハチ		
haq	→ ハッ		
kyuu	→ キュウ		
ku	→ ク		

keta-zyuu → zyuu  
keta-zyuq → zyuq  
keta-zyuu → ni zyuu  
keta-zyuq → ni zyuq

表 1: 接続表

	ジュウ	ジュツ	ヒヤク	ヒヤツ	ビヤク	ビヤツ	ビヤク	ビヤツ	セン	ゼン
イチ	0	0	0	0	0	0	0	0	0	0
イツ	0	0	0	0	0	0	0	0	1	0
ニ	1	1	1	1	0	0	0	0	1	0
サン	1	1	0	0	1	1	0	0	0	1
シン	1	1	0	0	0	0	0	0	0	1
ヨン	1	1	1	1	0	0	0	0	1	0
ゴ	1	1	1	1	0	0	0	0	1	0
ロク	1	1	0	0	0	0	0	0	1	0
ロツ	0	0	0	0	0	0	1	1	0	0
シチ	1	1	1	1	0	0	0	0	1	0
ナナ	1	1	1	1	0	0	0	0	1	0
ハチ	1	1	0	0	0	0	0	0	0	0
ハツ	0	0	0	0	0	0	1	1	1	0
キュウ	1	1	1	1	0	0	0	0	1	0
ク	0	0	0	0	0	0	0	0	0	0

keta-hyaku -> hyaku  
keta-hyaq -> hyaq  
keta-hyaku -> ni hyaku  
keta-hyaq -> ni hyaq  
keta-byaku -> san byaku  
keta-byaq -> san byaq  
:  
keta-sen -> sen  
keta-sen -> iq sen  
keta-sen -> ni sen  
keta-zen -> san zen  
:

n-num-zyuu -> keta-zyuu  
n-num-zyuu -> keta-zyuu n-num-ichi  
n-num-hyaku -> keta-hyaku  
n-num-hyaku -> keta-hyaku n-num-ichi  
n-num-hyaku -> keta-hyaku n-num-zyuu  
n-num-sen -> keta-sen  
n-num-sen -> keta-sen n-num-ichi  
n-num-sen -> keta-sen n-num-zyuu  
n-num-sen -> keta-sen n-num-hyaku

これに対して、MSLR法をベースにすると、音声的な制約を接続表で記述し、数詞の生成規則だけをCFGで記述することによって、CFG規則の数を大幅に減らすことができる。

### プレターミナル規則

num-ichi -> ichi      num-zyuu -> zyuu  
num-ichi -> iq        num-zyuu -> zyuq  
num-ichi -> ni        num-hyaku -> hyaku  
num-ichi -> san      num-hyaku -> hyaq  
num-ichi -> shi      num-hyaku -> byaku  
num-ichi -> yon      num-hyaku -> byaq  
num-ichi -> go        num-hyaku -> pyaku  
num-ichi -> roku     num-hyaku -> pyaq  
num-ichi -> roq      num-sen -> sen  
num-ichi -> shichi   num-sen -> zen  
num-ichi -> nana  
num-ichi -> hachi  
num-ichi -> kyuu  
num-ichi -> ku

### 数詞生成規則

keta-zyuu -> num-ichi num-zyuu  
keta-hyaku -> num-ichi num-hyaku  
keta-sen -> num-ichi num-sen  
  
n-num-ichi -> num-ichi

以上のように、接続表の記述とCFGの記述を分離することにより各々の記述が簡潔になる。二つの記述のもつ制約をLR表に統合し、解析アルゴリズムとしてはただ一つ、通常のGLR法を用意するだけでよいので処理も簡潔になる。

### 3.2 文末述語の規則

日本語の文末述語は、動詞（形容詞・形容動詞）・助動詞・補助動詞・終助詞の相互承接関係が厳密に定まっている。しかも、それらが「入れ子型構造」になっていて、後にある助動詞・補助動詞・終助詞がそれ以前のものを含んでいる。たとえば「クレジットカード番号を確認します」という文の述語「確認します」をCFGの枠組で記述すると、たとえば次のようになる。

n-sahen -> 確認  
aux-suru-si -> し  
auxstem-masu -> ま  
vinfl-masu-syusi -> す  
  
n-sahen-renyo -> n-sahen aux-suru-si  
aux-masu-syusi -> auxstem-masu vinfl-masu-syusi  
vaux-masu-syusi -> n-sahen-renyo aux-masu-syusi

以上のように、CFG規則だけで記述しようとする、助動詞や補助動詞のそれぞれに固有の品詞カテゴリを与え、それらの相互承接関係を詳細に記述しなければならない。そのうえ、文末述語はヴォイスやアスペクトやモダリティや敬語表現などの要素を含み、それらが組み合わせ数的に結合して、多様な表現形式を生み出す。そのため、文末述語を生成する規則はおびただしい数に

なる。たとえば、「確認します」の代わりに「確認させていただきますか」というより丁寧な表現を解析するには、次のような拡大された規則が必要となる。

```
n-sahen -> 確認
aux-suru-sa -> さ
auxstem-caus-seru -> セ
auxstem-te-idan -> ていただけ
auxstem-masu -> ま
vinfl-masu-mizen -> せ
aux-negt-syusi -> ん
aux-sfp -> か

n-sahen-mizen -> n-sahen aux-suru-sa
aux-caus-seru-renyo -> auxstem-caus-seru
vaux-caus-renyo -> n-sahen-mizen aux-caus-seru-renyo
aux-te-renyo -> auxstem-te-idan
vaux-te-renyo -> vaux-caus-renyo aux-te-renyo
aux-masu-mizen -> auxstem-masu vinfl-masu-mizen
vaux-masu-mizen -> vaux-te-renyo aux-masu-mizen
vaux-negt-syusi -> vaux-masu-mizen aux-negt-syusi
vaux-sfp -> vaux-negt-syusi aux-sfp
```

このように、文末述語の表現が増えれば増えるほど、それを解析するための規則もまた際限なく増え続けることになる。事実、これまでに開発してきた文法規則の総数 1809 のうち、述語を生成する規則数は 920 にのぼる。

しかし、MSLR 法をベースにすれば、こうした労力が著しく軽減される。それによると、同じ文末述語を記述するのに、次のような語彙規則と句構造規則、両者のインターフェースとなるプレターミナル規則、そして、動詞・助動詞・補助動詞・終助詞の相互承接関係を記述した接続表を用いる。

#### 語彙規則

```
n-sahen -> 確認
auxv-suru-sa -> さ
auxstem-caus-seru -> セ
auxstem-te-idan -> ていただけ
auxstem-masu -> ま
vinfl-masu-syusi -> せ
auxstem-negt-syusi -> ん
aux-sfp-ka -> か
```

#### 句構造規則

```
verb -> n-sahen auxv
vaux -> verb aux
vaux -> vaux aux
aux -> auxstem
aux -> auxstem vinfl
aux -> aux-sfp-ka
```

#### プレターミナル規則

```
n-sahen -> sahen-meisi
auxv -> auxv-suru-sa
auxstem -> auxstem-caus-seru
auxstem -> auxstem-te-idan
auxstem -> auxstem-masu
vinfl -> vinfl-masu-syusi
auxstem -> auxstem-negt-syusi aux-sfp -> aux-sfp-ka
```

#### 接続表

```
sahen-meisi : auxv-suru-sa
auxv-suru-sa : auxstem-caus-seru
auxstem-caus-seru : auxstem-te-idan
auxstem-te-idan : auxstem-masu
auxstem-masu : vinfl-masu-mizen
vinfl-masu-mizen : auxstem-negt-syusi
auxstem-negt-syusi : aux-sfp-ka
```

注 この表は、「:」の左辺のプレターミナルに右辺のプレターミナルが接続可能であるということを表している。

「入れ子型構造」は句構造規則を再帰的に適用することで実現されていることに注意してほしい。文末述語は、助動詞・補助動詞・終助詞のどのような承接であっても、基本的にはこれら少数の規則によって解析することができるのである。事実、このようにして作成された述語関連の規則数はたかだか 62 に過ぎない。前述のような、接続表の制約も CFG 規則で記述する文法において述語関連の規則数は 920 であるから、MSLR 法をベースにした文法がいかに簡明で見通しのよいものであるかわかるだろう。

#### 4 文法の比較

MSLR 法をベースにした文法（新 CFG）と、接続表の制約をもすべて CFG 規則として記述した文法（旧 CFG）とを、規則数、および単語パーブレキシティで比較してみる。両文法が対象としている文は、ATR 音声言語データベースから選んだ 12 対話である。ただし MSLR 法で用いた接続表は、この 12 対話 + 50 対話から自動的に作成したものを用いる。

表 2 によれば、旧文法の規則数 1809 に対して、新文法の規則数は 628 個であった。ただし新文法では、辞書より自動的に生成可能なプレターミナル規則数が 327 個を占めているので、人手で開発した純粋な句構造規則は 301 個である。したがって、文法開発に要す労力の観点から比較すると、1/6 に減少している。

ほぼ同等の能力を持つと考えられる両文法の単語パーブレキシティを比較すると、旧文法が 71.2 に対して、新文法は 31.0 で、45% 以下に減少している。

単語パーブレキシティをさらに減少させる方法を、竹澤らは開発している [竹澤 96]。旧文法にプレターミナルのバイグラムデータを加えた言語モデルのパーブレキシティは 27.5 である。新文法と旧文法のプレターミナルのセットが異なること、新文法には接続表の制約が記述されていないが、旧文法には接続表の制約が記述されているために、竹澤らの方法を新文法に対してそのまま適用して比較するわけにはいかない。プレターミナルのバイグラムデータを LR 表に組み込む方法を、[Li 96] が開発している。この方法は、接続表の制約を組み込み済みの LR 表のアクションに、バイグラムの確率を振る方

表 2: 規則数およびパープレキシティの比較

言語モデル	語彙サイズ	規則数	パープレキシティ
旧文法	1010	1809	71.2
新文法	1010	628	31.0

法である。詳細は別稿に譲るが、この方法を用いて MSLR 法をベースにした新文法による単語パープレキシティと、竹澤らの方法によるものとを比較する実験を今後行なってみることは意味があるだろう。ここでバイグラムは、接続表の各エントリーに書き込まれている、接続可能、不可能を表す 1 と 0 の値の代わりに、確率値を用いたものに相当していることに注意したい。われわれは、バイグラムではなく、確率 LR の新しい理論を構築しつつある。これは Briscoe らの理論、方法 [Briscoe93] の不備を補い完全にしたものである。Li の方法は、その近似になっている。その詳細は機会を改めて発表したい。

## 5 まとめ

MSLR 法と呼ぶ自然言語処理の新しい方法を提案した。それによれば、CFG 規則と接続表を別個に記述し両方を LR 表に組み込むことで、自然言語の多様な表現を簡潔に、しかも精密に記述できることを実証した。

この MSLR 法を音声認識用日本語文法に応用し、CFG 規則のみによる文法と比べて、規則数が大幅に減少し、文法が見通しよく記述可能となり、保守や文法体系の拡張が容易になることを報告した。

統計データを利用する方法として、確率 CFG [Fujisaki91]、ルールバイグラム [Kita94]、Li の方法 [Li 96]、竹澤の方法 [竹澤 96] などがある。これらの言語モデルについても、パープレキシティの観点から、今後検討を行なう予定である。

最後に、われわれは複数の接続表の制約を LR 表に組み込む方法を開発している [綾部 97]。それによれば、たとえば異音間の接続表とプレターミナル間の接続表の制約を組み込んだ LR 表を用いることにより、二つの異なるレベルの接続表の制約を同時に用いた音声認識システムを構築することが可能になる。今後、このような実験を行ないたいと考えている。

## 参考文献

[綾部 97] 綾部寿樹, 徳永健伸, 田中穂積: 複数の接続表の LR 表への組み込み -LR 表工学 (2). 情報処理学会自然言語処理研究会, (1997 年 1 月発表予定)

[Briscoe93] Briscoe, T. Carrol, J.: Generalized Probabilistic LR Parsing of Natural Language (corpora) with Unification-based Grammars. Compu-

tational Linguistics, 19,1, pp.25-59(1993)

[Fujisaki91] Fujisaki, T. Jelinek, F. Cocke, J. Black, E. Nishino, T.: A Probabilistic Parsing Method for Sentence Disambiguation. in [Tomita91]

[Kita94] Kita, K. et al.: Spoken Sentence Recognition Based on HMM-LR with Hybrid Language Modeling. IEICE Trans. Inf. & Syst., Vol. E77-D, No2(1994)

[Knuth65] Knuth, D. E.: On the Transition of Languages from Left to Right. Information and Control, 9, 607-639 (1965)

[Li 96] Li, H.: Integrating Connection Constraints into a GLR Parser and its Applications in a Continuous Speech Recognition System. TR96-0003, Dept. of Computer Science, Tokyo Institute of Technology(1996)

[竹澤 95] 竹澤寿幸, 田代敏久, 森元逞: 自然発話の言語現象と音声認識用日本語文法, 情報研報, 95-SLP-6-5 (1995)

[竹澤 96] 竹澤寿幸, 森元逞: 部分木に基づく構文規則と前終端記号バイグラムを併用する対話音声認識手法, 信学論, D-II, Vol. J79-D-II, No. 12, pp. 2078-2085 (1996)

[田中 89] 田中穂積: 自然言語解析の基礎, 産業図書 (1989)

[田中 95] 田中穂積: 自然言語解析の新しい方法 - LR 表工学の提案 (1), 人工知能学会研究会資料, SLG-J-9501-1, pp. 1-8 (1995)

[Tomita86] Tomita, M.: Efficient Parsing for Natural Language: A Fast Algorithm for Practical Systems. Kluwer Academic Publishers, Boston, MA(1986)

[Tomita91] Tomita, M.(ed.): Generalized LR Parsing Technologies. Kluwer Academic Publishers (1991)

[植木 95] 植木正裕, 徳永健伸, 田中穂積: EDR 辞書を用いて日本語の形態素解析と統語解析を行なうシステム. EDR 電子化辞書利用シンポジウム, pp.33-39(1995)