構文構造のグループ化による漸進的チャート解析の高速化

浅井 悟† 松原 茂樹‡ 外山 勝彦† 稲垣 康善†

†名古屋大学大学院工学研究科 ‡名古屋大学大学言語文化部

{asai,matu,toyama,inagaki}@inagaki.nuie.nagoya-u.ac.jp

1 はじめに

実時間話し言葉処理システムの実現を目指し, 著者らはこれまでに漸進的チャート解析手法を提案している [2]. この手法は, 上昇型解析と下降型解析を融合することにより実現されており, 語が入力されるたびに, それまでの入力に対する構文構造を少なくとも一つ作り上げることができる.

しかし, 下降型解析を導入したことにより, 解析途中における未入力部分に関して, より多くの予測を行うことになるため, 通常の上昇型チャート解析に比べ, 作成される構文構造が多くなり, 解析効率が低下する. そのような問題の解消のためには, 解析を進める上での共通の性質を備えた構造をひとまとめにして処理し, 記憶量を圧縮することは一つの方法である [3].

そこで本稿では, 解析途中で作成される構文構造をグループ化することにより, 漸進的チャート解析を効率化する手法を提案する. この手法では, 個々の構文構造に対して実行していた処理を, グループ単位で一括して行えるため, 必要な処理量が削減され, 解析の高速化が期待できる. また本手法では, 入力全体の構文構造を作り上げることなく, 分解した状態でそれを保持するするため, 処理を進める上での記憶容量を節約することができる.

2 漸進的チャート解析とその問題点

2.1 漸進的チャート解析

漸進的チャート解析は, チャート解析 [1] を基にした, 文脈自由文法に対する構文解析手法の一つである. 漸進的チャート解析では, 構文解析の途中経過を表現するために, **チャート**と呼ばれるラベルつきグラフを用いる. チャートの**節点**は, 入力文の単語と単語の間に付けられた番号であり, 節点間に張られる**弧**には, **項**と呼ばれる構文構造がラベルとして付加される. 例えば, "*Ken*" に対して図1の辞書 ($d1$) を用いると, 節点0, 1の間に弧が張られ, 項 $[Ken]_{\mathrm{NP}}$ がラベルとして付加される.

α を範疇とするとき, 項 $[?]_\alpha$ を**未決定項**と呼ぶ. 特に, 項の中の最も左に位置する未決定項を**最左未決定項**という. 未決定項を含む項をラベルにもつ弧を**活性弧**と呼び, そうでない弧を**不活性弧**と呼ぶ.

漸進的チャート解析では, 無駄な辞書引きや文法規則の適用を抑制するため, **到達可能性**と呼ばれる範疇間の関係を利用する. 範疇 α が範疇 β に到達可能であるとは, 直観的には, α を範疇とする項が β を範疇とする項の左端に位置し得ることを意味する. 以下では, 範疇 α が範疇 β に到達可能であることを $\alpha \rightsquigarrow \beta$ で表す.

漸進的チャート解析では, 項 $[?]_S$ をもつ弧が節点0, 0の間に張られたチャートを初期状態として, i 番目の語 w_i に対して以下の手続き a) ~ c) を順に実行する. なお, 弧のラベルが項 t であるとき, t をその弧の項と呼ぶ.

a) 辞書引き 節点 $0, i$–1 の間に張られた弧の項を t, その範疇を β とする. w_i の範疇 α が $\alpha \rightsquigarrow \beta$ ならば, 項 $[w_i]_\alpha$ をラベルにもつ弧を節点 i–1, i の間に張る.

b) 文法規則の適用 節点 i–1, i の間に張られた弧の項を t', その範疇を σ_1 とする. このとき, 文法規則 $\gamma \rightarrow \sigma_1\sigma_2\cdots\sigma_n$ が存在し, かつ, 節点 0, i–1 の間に張られた弧の項 t の最左未決定項 $[?]_\beta$ に対して $\gamma \rightsquigarrow \beta$ ならば, 項 $[t'\ [?]_{\sigma_2}\cdots[?]_{\sigma_n}]_\gamma$ をラベルにもつ弧を節点 i–1, i の間に張る. この操作を可能な限り繰り返す.

c) 項の置き換え 節点 0, i–1 の間に張られた弧の項 t の最左未決定項を $[?]_\delta$, 節点 i–1, i の間に張られた弧の項を t' とする. このとき, t' の範疇が δ ならば, $[?]_\delta$ を t' で置き換えた項をラベルにもつ弧を節点 0, i の間に張る.

漸進的チャート解析は, 上昇型としての側面と下降型としての側面を併せもった構文解析アルゴリズムであるといえる. 手続き a), b) は, 上昇型解析に相当し, 手続き c) は, 下降型解析に相当する. これにより, それまでの入力に対する構文構造を任意の時点で作成することが可能となる.

漸進的なチャート解析の例として, 入力文

(2.1) *Ken met Mary at the station.*

における語 "*met*" が入力された時点での置き換え操作を図3(置き換え操作に直接関係する弧のみを示す) を用いて説明する. "*met*" が入力された時点では, 既に節点 0, 1 の間に2つの弧が張られており, 語 "*Ken*" までの入力に対する項 (1), (2) がそれぞれラベル付けられている. このとき, 手続き a), b) を実行することにより, 語 "*met*" に対して項 (3) をラベルにもつ弧が節点 1, 2 の間に張られる. 項 (1) の最左未決定項を項 (3) で置き換えることにより, 項 (4) をラベルにもつ弧が節点 0, 2 の間に張られる. 同様にして, 項 (5) をラベルにもつ弧が節点 0, 2 の間に張られる. 項 (4), (5) はいずれも, "*Ken met*" に対する構文構造を表している.

2.2 漸進的チャート解析の効率と記憶量

前節で説明したように, 漸進的チャート解析では, 語が入力されるたびにそれまでの入力に対する項を作り上げることができる. しかし, この手法にはいくつかの問題がある.

まず, (1),(2) はいずれも"*Ken*" の入力に対して作成された項であり, その最左未決定項の範疇は VP である. そのため, どちらの項も範疇 VP の項 (3) を用いて置き換え操作が行われるが, 同じ操作を繰り返すことになるため, 効率が悪い. また, その操作によって作成された項 (4),(5) は, いずれ

(d1) NP → *Ken*
(d2) V → *met*
(d3) NP → *Mary*
(d4) Prep → *at*
(d5) Det → *the*
(d6) N → *station*

図 1: 辞書

(g1) S → NP VP
(g2) S → NP VP PP
(g3) VP → V NP
(g4) PP → Prep NP
(g5) NP → Det N

図 2: 文法

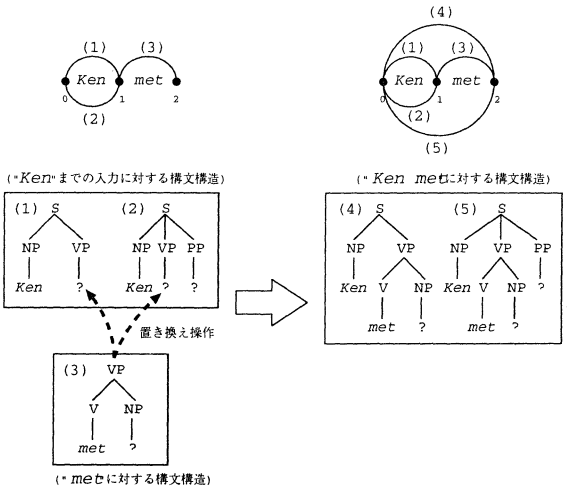

図 3: "*met*" が入力された時点における置き換え操作

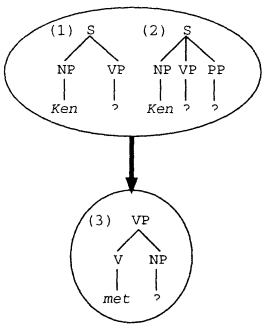

図 4: 項 (1), (2) と項 (3) の関係を表す有向グラフ

も共通の項 (3) を含むことになるが, それを別々に記憶するため, 記憶量に関して無駄が大きい. さらに, 通常の上昇型チャート解析であれば, 活性弧の項の最左未決定項を別の活性弧の項で置き換えることはないため, "*met*" が入力された時点で (4),(5) が作成されることはない. よって, 次の入力語 "*Mary*" に対して, 例えば, 項 $[Mary]_{\mathrm{NP}}$ が作成された場合でも, それを用いて項 (3) を置き換える操作を行えばよい. それに対して漸進的チャート解析では, 作成された項 (4),(5) のいずれに対しても項 $[Mary]_{\mathrm{NP}}$ を用いた置き換え操作を行う, すなわち, 同じ操作を繰り返すことになるため, 効率が悪い.

すなわち, 漸進的チャート解析の問題は以下のようにまとめられる.

(A) 同じ最左未決定項をもつ複数の項は, 以後の入力に対して同じ項を用いた置き換え操作が行われる (処理効率の問題).

(B) さらに, そのような置き換え操作により, 同一の項が複数の異なる項の中で別々に記憶される (記憶量の問題).

(C) 複数の項に対して同一の項で置き換えを行うと, 複数の項が新たに作成されるが, それらは以後, いずれも同じ項との間で置き換え操作が行われる (処理効率の問題).

これらの問題は, 一般に, 入力文が長くなるにつれて, また, 文法規則の数が増えるにつれてより深刻となる.

3 項のグループ化に基づく漸進的チャート解析

前節で述べた問題を解消するために, 漸進的チャート解析を以下のように改良する.

まず, 問題 (A) を解消するために, 置き換えを行う上で共通の性質を備えた項をひとまとめにして扱う. ある項が他の項との間で置き換え操作を行えるかどうかは, その項の範疇とその最左未決定項の範疇に依存する. そこで, それらの対が等しい項をグループ化し, グループ間の置き換え可能性を図る. 図 3 の場合, 項 (1),(2) からなるグループを構成することができ, それに対して項 (3) を用いて置き換えを図ることにより, 処理の効率化が可能となる.

また, 問題 (B) 及び (C) を解消するために, 実際には置き換え操作を実行しない. これにより, 記憶量の節約が可能となるとともに, 置き換え可能な項の組合せの数が減るため, 処理を効率化できる. 図 3 の場合, 項 (1),(2) と項 (3) との間でそれぞれ置き換えをしなければ, 新たに項 (4),(5) を記憶する必要はない. また, 語 *Mary* に対する項 $[Mary]_{\mathrm{NP}}$ については, 項 (3) に対してのみ置き換えを行えばよい.

なお, あらかじめ置き換え可能なグループの間の関係を示す指標をグループに付加すれば, 必要なときにいつでも, 全体の構文構造を表す項を作り上げることができる.

以下では, グループ及びそれに付加されるラベルについて説明し, 改良後の漸進的チャート解析アルゴリズムを示す.

3.1 項のグループ化

入力された語に対する辞書引き及び文法規則の適用によって作成される項を, その範疇と最左未決定項の範疇をもとにまとめたものを項の**グループ**と呼ぶ.

あるグループに属する項の最左未決定項が, 別のグループの項で置き換え可能であることを表現するために, 有向グラフを用いる. 節点はグループを表し, 節点間を結ぶ有向辺は, 始点のグループの項の最左未決定項を終点のグループの項で置き換え可能であることを表す. 例えば, 同じ最左未決定項をもつ項 (1), (2) と, それらを置き換え可能な項 (3) との関係は, 有向グラフを用いて図 4 のように表せる. 図 4 において, 楕円は一つのグループに対応する. 以下では, グループ G から G' への有向辺を**リンク**と呼び, G を**リンク元グループ**, G' を**リンク先グループ**と呼ぶ. リンクをたどって置き換え操作を行うことにより, ある項を別のある項で置き換えた項を得ることができる.

また, 項のグループ化とリンクを用いることにより, 複数のグループが 1 つのグループを共有することができる. グループの共有化により, 記憶領域を節約できる.

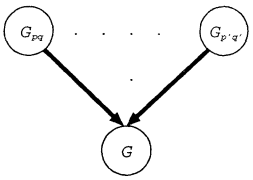

図 5: G のリンク元グループ

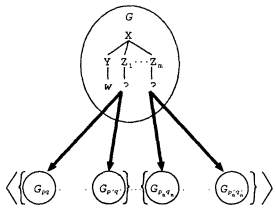

図 6: G のリンク先グループリスト

3.2 グループのラベル

グループ及びそれに属する項の情報を表現するため, グループにラベルを付加する. グループのラベルは 6 項組 $[G, C, L, k, F, T]$ で表され, その要素はそれぞれ次のような意味をもつ.

- グループ名 G
  G はグループ名を表す. i 番目の語が入力された時点で j 番目に作成されたグループを G_{ij} と表記する.
- 範疇 C
  グループ G に含まれる項の範疇を表す.
- 最左未決定項の集合 L
  グループ G の最左未決定項の集合を表す. ただし, L がリンク先グループ名の集合であるとき, G の最左未決定項はリンク先グループの最左未決定項である.
- 解析節点番号 k
  グループ G に含まれる項が, 語 w_k までの入力に対する項の構成要素の一部であることを表す.
  解析節点番号は, 3.3 節で説明するフィルタリングに用いる.
- リンク元グループの集合 F
  F はグループ名の集合であり, G が F に属するグループのリンク先グループであることを表す. 例えば, $F = \{G_{pq}, \cdots, G_{p'q'}\}$ であるとき, グループ G との関係を, 図 5 のように表すことができる. なお, グループ G_{0j} $(j > 0)$ のリンク元グループの集合を, $\{root\}$ と表記する.
- リンク先グループの集合のリスト T
  T はグループ名の集合のリストであり, グループ G が T に属するグループのリンク元グループの集合であることを表す. 例えば, $T = \langle \{G_{p_1q_1}, \cdots, G_{p'_1q'_1}\}, \cdots, \{G_{p_nq_n}, \cdots, G_{p'_nq'_n}\} \rangle$ であるとき, グループ G との関係を, 図 6 のように表すことができる.

入力： 長さ n の自然言語文 $w_1 w_2 \cdots w_n$

出力： $w_1 w_2 \cdots w_n$ に対する有向グラフ

手順：

(初期化) $i := 0$. 節点 0, 0 の間に弧を張り, 項 $[?]_S$ をラベルとして付加する. この項のみからなるグループを G_{00} の要素とし, そのラベルを $[\, G_{00}, S, \{S\}, 0, \{root\}, \langle\{\}\rangle \,]$ とする.

(Step1) (辞書引き)
$i := i + 1$. i 番目の語 w_i を読み込み, w_i を辞書引きした項をラベルにもつ弧をチャートに加える.

(Step2) (文法適用)
節点 i–1, i の間に張られた弧の項を t', その範疇を σ_1 とする. このとき, 文法規則 $\gamma \rightarrow \sigma_1 \sigma_2 \cdots \sigma_n$ が存在し, かつ, 節点 0, i–1 の間に張られた弧の項 t の最左未決定項 $[?]_\beta$ に対して $\gamma \rightsquigarrow \beta$ ならば, 項 $[t'\ [?]_{\sigma_2} \cdots [?]_{\sigma_n}]_\gamma$ をラベルにもつ弧を節点 i–1, i の間に張る. この操作を可能な限り繰り返す.

(Step3) (グループ化)
G_{pq} $(p < i)$ の最左未決定項の集合を $\{[?]_r\}$ とする. このとき, 節点 i–1, i の間に張られた活性弧の項で, その範疇が r, 最左未決定項が l である項からなるグループを G_{ij} とし, そのラベルを $[\, G_{ij}, r, \{l\}, i, \{\}, \langle\{\}\rangle \,]$ とする. また, 不活性弧の項で, その範疇が r である項からなるグループのラベルを $[\, G_{ij}, r, \{\}, i, \{\}, \langle\rangle \,]$ とする.

(Step4) (グループ間の関連づけ)

(Step4-1) (リンク)
G_{pq} の最左未決定項の集合を $\{[?]_r\}$ とする. このとき, 範疇が r であるグループ G_{ij} を G_{pq} におけるリンク先グループリストの最後の要素に加える. さらに, G_{ij} のリンク元グループの集合に G_{pq} を加える.

(Step4-2) (解析節点番号の更新)
G_{ij} のリンク元グループの集合のみを利用してたどることのできるすべてのグループの解析節点番号を i に更新する. 更新したグループのリンク先グループのみを利用してたどることのできるすべてのグループの解析節点番号を i に更新する.

(Step4-3) (最左未決定項の集合の更新)
G_{ij} と最左未決定項の集合が $\{\}$ であるグループに印をつける. G_{ij} 以外のすべてのグループの最左未決定項の集合を $\{\}$ にする. 印がついていない任意のグループ G_{pq} $(p < i)$ に対して, その最左未決定項の集合をリンク先グループリストの最後の要素に変更し, 印をつける.

(Step5) (フィルタリング)
グループの解析節点番号が i 未満であるすべてのグループを削除する. **Step1** に戻る.

図 7: グループ化を用いた漸進的チャート解析アルゴリズム

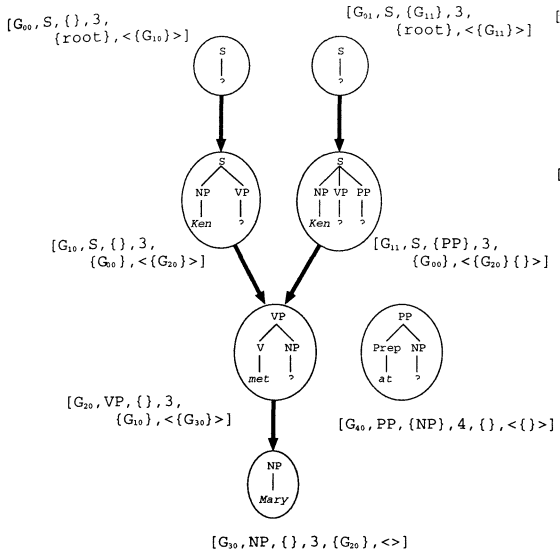

図 8: "*at*" に対する解析処理 (**Step3** 終了後のグラフ)

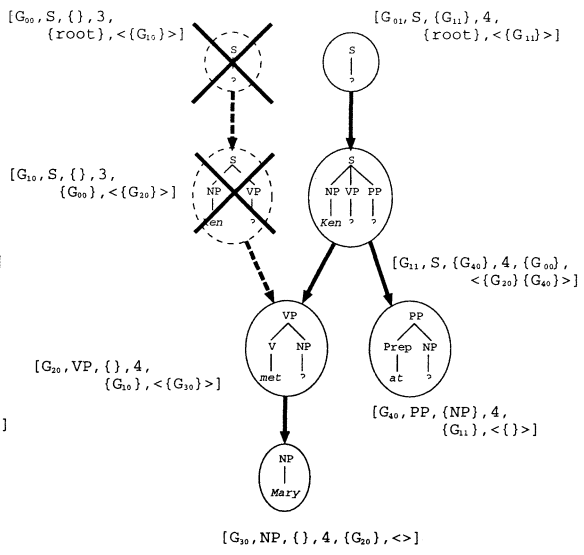

図 9: "*at*" に対する解析処理 (**Step5** 終了後のグラフ)

3.3 グループ化を用いた漸進的チャート解析アルゴリズム

グループ化を用いた漸進的チャート解析アルゴリズムを図 7 に示す. **Step5** では, それ以降の入力に対する解析処理に使用しないグループをグラフから削除する. この操作をグループの**フィルタリング**と呼ぶ. フィルタリングを行うことにより, 項を記憶するために必要な記憶領域を節約することができる.

このアルゴリズムを用いれば, それまでの入力に対する構文構造をグラフで表現できる. また, リンクをたどることにより, それまでの入力に対する構文構造を得ることができる.

3.4 解析例

提案手法を用いて, 入力文 **(2.1)** を解析する. 辞書と文法は図 1, 図 2 に示すものを使用する. 以下では, 語 "*at*" が入力され, **Step3** まで解析が進んでいるものとする. この時点でのグラフを図 9 に示す.

ここで, **Step4-1** を実行すると, G_{11} のラベルのリンク先グループの集合のリストは $\langle\{G_{20}\}\{G_{40}\}\rangle$ に変更される. これは, G_{11} から G_{40} にリンクが張られたことを意味する. 次に **Step4-2** を実行すると, G_{01}, G_{11}, G_{20}, G_{30} の解析節点番号はどれも 4 に変更され, さらに, **Step4-3** を実行することにより, G_{11} の最左未決定項の集合が, リンク先グループ名である $\{G_{40}\}$ に変更される. **Step5** を実行することにより, G_{00} と G_{10} は削除される. **Step5** が終了した時点でのグラフを図 10 に示す. G_{00} と G_{10} は, フィルタリングによって削除されたことを表している.

$\{root\}$ をリンク元グループの集合の値としてもつグループからリンクをたどり, 置き換え操作を実行することにより, "*Ken met Mary at*" に対する構文構造が得られる.

4 おわりに

本稿では, 漸進的チャート解析において, 解析途中で作成される項を, その範疇と最左未決定項の範疇をもとにグループ化することにより, 効率的に解析可能な手法を提案した. 項のグループ化により, 置き換え可能な項の組合せの数が減るため, 全体として処理の高速化が期待できる. また, グループ間の関係を有向グラフとして表現することにより, 複数のグループで一つのグループを共有することが可能になった. これにより, 記憶量の圧縮を実現した.

本手法を計算機上に実装し, 解析実験を通して, 本手法を評価することは, 今後の課題である.

参考文献

[1] Kay, M.: Algorithm Schemata and Data Structures in Syntactic Processing, *Technical Report CSL-80-12*, Xerox PARC (1980).

[2] Matsubara, S., Asai, S., Toyama, K. and Inagaki, Y.: Chart-based Parsing and Transfer in Incremental Spoken Language Translation, *Proc. of the 4th Natural Language Processing Pacific Rim Symposium*, pp. 521-524 (1997).

[3] 田中 穂積: 自然言語解析の基礎, 産業図書 (1989).