長い日本語文における並列構造の推定†

黒橋禎夫†† 長尾 眞††

日本語情報処理において未解決のまま放置されている問題の1つに，長い文を正しく構文解析することがほとんどできないという問題がある．文が長くなる主な原因は，1文中に多くの内容が並列的に述べられているところにある．したがって，このような並列する構造を正しく認識できれば，長い文も短くすることができ，文の解析が正しくできる可能性が高くなる．多くの文において，並列する部分は何らかの意味において類似している．そこで，文中の並列構造を類似した2つの文節列としてとらえ，これをダイナミックプログラミングの手法によって発見することを実現した．並列構造としては，名詞句の並列のほかに，いわゆる連用中止法といわれている述語句の並列等を対象とした．まず，日本語文を文節ごとに区切り，すべての文節対について類似度を計算する．そして，並列の存在を示す助詞や連用中止などの前後において，バランスのとれた並列構造を優先すること，文を意味的に区切っているある種の表現をこえて並列の範囲が広がる可能性は少ないこと，並列構造の直後に「など」のような語が現れやすいこと，等を考慮に入れた上で，類似度の総和が最も大きい2つの文節列を求め，これを並列構造の範囲とする方法を考案した．180文に対して実験を行ったところ，この方法によって82%の精度で並列構造を推定することができた．

1. はじめに

機械翻訳を代表とする日本語情報処理は最近かなりの進展をみせている．しかし困難な問題はいくつも残されており，その1つに長い文の解析の問題がある．日英機械翻訳システムは数多く出され実用されはじめているが，漢字かなまじり文で50文字以上の文の解析は非常に困難であり，80文字以上の文の解析はほとんどが失敗するといわれている．

そこでその失敗の原因を考えてみると，「長い文の場合，1つの文節の係り先がいくつもありうるから」ということが主たる原因と考えられるが，それがなぜ解析の失敗につながるかということが明らかでない．ただ，膨大な数の曖昧さが生じる原因の1つは採用している文法形式にあると考えられる．今日まで言語の文法といえば，多くの場合チョムスキーの提案した句構造文法にたよってきた．また属性をもたせた拡張句構造文法がいろいろと提案され，ユニフィケーションという操作を導入することによって種々の試みが行われてきている．また古くから依存構造文法（係り受け解析）があり，一方では格文法という考えが導入され，語順にあまり影響されない解析が行えるようにもなっている．解析結果も句構造表現，依存構造表現，格構造表現，その他種々の形式が存在する．しかしこれらいずれの場合にも基本的な考え方は近くの少数の要素の関係をしらべるということの繰り返しによっている．

しかし言語表現はそんなに単純なものではなく，いくつもの，しかもかなり離れたところにある要素同士が呼応しあうということがよくある．構文解析によって多くの可能な解析結果を出した（出せる）といったことは，ある意味では文法が不完全だから排除すべき構造まで出してしまうわけで，そのように多くの解析結果が出るのは，隣り合う2要素の関係しか見ないところに主たる原因があるといえよう．

このような問題を解決する方法の1つは，各要素に付属させる（意味）属性値をできるだけ精密なものにし，適切な規則だけが適用されるようにすることであろう[1]．他は文中に広くひそかに存在する多くの要素同士の関係を同時的に検出することである．文が長くなる主な原因は，1文中に多くの内容を並列的に述べようとするところにある．特に科学技術文章などにおける長い文ではそういったことが多い．並列的なものとして，ここでは並列名詞句，並列形容詞句などのほかに並列する文，すなわち連用中止法による文の継続的接続も含めて考えている．したがって，このような並列する構造を正しく認識できれば，長い文も短くすることができ，文の解析が正しくできる可能性が高くなる．

従来から並列構造の解析については種々の研究がなされ[2),3)]，文法規則にも工夫がなされてきたが，あまり成功しているとはいえない．したがって，これまで

† A Method for Analyzing Conjunctive Structures in Japanese by SADAO KUROHASHI and MAKOTO NAGAO (Department of Electrical Engineering, Faculty of Engineering, Kyoto University).

†† 京都大学工学部電気工学第二教室

の方法とはかなり違った考え方を導入しないと，並列構造の把握に成功することは難しい．

1文中において並列する語句は何らかの意味において類似しているということが多い．したがって，長い文の中に存在する類似した2つの単語列（日本語の場合，文節列）を従来の文法規則の適用といったこととは別の方法で発見することができる必要があり，それができると並列構造の解析として役に立つわけである．本論文は，これを音声認識などで広く使われているダイナミックプログラミングによるマッチング法(DP マッチング）と同様の考え方によって実現した．

2. 日本語文における並列構造

まず日本語文における並列構造にどのようなものがあるかを整理しておく．その詳細は3.2節で述べる．

1つには，「と」や「,(読点)」などによって接続される名詞の並列がある．

(i)　…解析と 生成を….

(ii)　…原言語の 解析と 相手言語の生成を….

(iii)　…原言語を解析する 処理と 相手言語を生成する処理を….

これには上の例文のように，単独名詞の並列，修飾語を伴う名詞の並列，連体修飾節を伴う名詞の並列がある．ここでは，これらをすべて含めて**名詞並列**とよぶことにする．

もう1つは，1文内に複数の述語があり，それらが並列的に，すなわちどちらが主とも従ともいいがたい平等の関係でつながっているという構造がある*1．

(i)　…原言語を 解析し 相手言語を生成する（処理を…）．

(ii)　…解析では 利用するが，生成では利用しない（という…）．

この種の並列の存在は(i)のように活用形によって示される場合と，(ii)のように接続助詞などによって示される場合がある．このような並列構造を**述語並列**とよぶことにする．

その他の並列構造として，述語並列から述語を除いたある一部分が並列的につながっているというものがある．

(i)　…前者を 解析に，後者を生成に….

(ii)　… 解析に，または 生成に….

これらには(i)のように助詞が呼応している場合と，(ii)のように明示的に並列構造を示す表現（「または」など）を伴う場合がある．このような並列構造を**部分並列**とよぶことにする．

また，各並列構造の存在を示す表現（上の各例文の下線部分）を**並列のキー**とよぶことにする．

3. 並列構造の推定の方法

本論文では並列構造の推定をつぎのような問題として扱う．

並列のキーに対して，その前後のどの範囲が並列構造をなす2つの文節列であるかを推定する．

例えば，「…AのBとCのDを…」という表現において，「と」によって 'B' と 'C' の並列が示されているのか，'AのB' と 'CのD' なのか*2，あるいは前後のさらに広い範囲までを含むのかを推定することになる．

この方法の概要を，**図1の三角行列**によって説明する．

前処理　まず入力日本語文を形態素解析し，自立語とそれに続く付属語を1つのブロックにまとめる（以後，この各ブロックを**文節**とよぶ）．図の各対角要素が1つの文節である．

並列のキーの発見　次に，並列構造の存在を示す表現を発見し，並列のキーとする．図では，'a>'*3 のついている文節が並列のキーであることを示している．

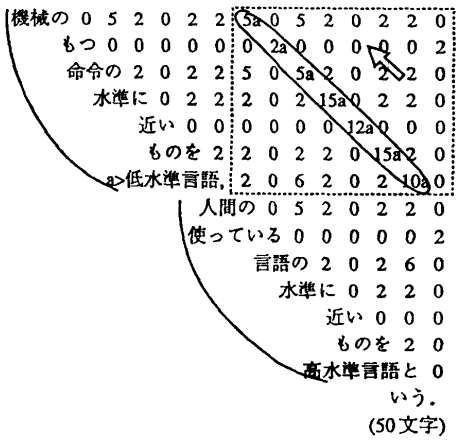


図1　並列構造の推定の例
Fig. 1　An example of analyzing conjunctive structures.

*1 これに対して，主従関係で述語が接続されている場合（因果関係を示す「～するので，～」など）は，主節と従属節の間に類似性が存在するとは限らないため，本手法の解析対象とはしない．

*2 'B' と 'CのD', 'AのB' と 'C' なども考えられる．

*3 1文中に複数の並列のキーが存在する場合もあるので，これらをアルファベットを添えて区別している．

文節間の類似度の計算 すべての文節の対について類似度を計算する．類似度としては自立語が同じ品詞である場合，同じ付属語を含む場合などにポイントを与える．三角行列の (i, j) 要素の数字は i 番目と j 番目の文節の類似度のポイントを示す．

並列構造の範囲の推定 並列のキーの前後で，類似度の総和が最も大きい文節列の組を求める．これは，図の点線の範囲内において，一番下の行の1つの要素から出発して点線の範囲内の一番左の要素までの左上方向への要素の並び（以後，これを**パス**とよぶ）の中でポイントの和[*4]が最も大きいパスを求めることに対応する．この処理はDPマッチングの手法で行う．そのようなパスが求まれば，そのパスの左側の対応する文節列と下側の対応する文節列が並列であると推定する．図の例文では，添字 'a' のついたパスが最高得点 64 を得るパスで，このパスに対応する2つの文節列，「機械の〜低水準言語，」と「人間の〜高水準言語と」が並列であると推定される．この処理について以下で詳しく説明する．

3.1 前処理

まず形態素解析[*5]によって入力日本語文を単語単位に分割し，各単語の品詞，さらに活用する語の場合には活用形と原形を決定する．この解析は複数の解析結果を出すが，並列構造の推定に関心があるので複合語などはなるべく一単語として扱うほうがよいという理由から，以下の処理はこのうち単語数最少，自立語数最少の解に対して行う．

次に分割された単語列を文節（自立語とそれに続く付属語）にまとめる．ここでは，読点も付属語とみなし，読点の次に「または」，「あるいは」などの接続詞が続く場合にはそれらも同一文節の付属語とみなす．また，意味的まとまりという観点から「サ変名詞＋する」，「サ変名詞＋を＋行う」などは1つの文節にまとめ，自立語の品詞は動詞であるとし，「する」，「行う」などは付属語とみなす．

3.2 並列のキーの発見

2章で分類した各並列構造に対して，次のような条件を満たす文節を並列のキーとする．

名詞並列 自立語が名詞であり，かつ**表1**に示す語を付属語に持つ文節を並列のキーとする．ただし「副詞的名詞（今後，通常，など）＋読点」や，「〜と」の直後の文節が「いう」，「思う」，「比較する」などの場合は並列のキーとしない．また，付属語「も」を持つ文節については，その文節より後ろの文節の中に付属語「も」を持つものがある場合（「〜も〜も」のような場合）にのみ並列のキーとする．

なお「・(中点)」はほとんどの場合，その前後の単独名詞同士の並列を示していると解釈できるので，並列のキーとはしない．

述語並列 自立語が用言であるか付属語に判定詞[*6]がある場合にのみ次のことを調べる．

文節の活用形[*7]が連用形[*8]であり，かつ付属語に読点がある文節を並列のキーとする．読点を伴わない連用中止はほとんどの場合次の述語に結びつくと考えられるので並列のキーとはしない．また，表1に示す語を付属語に持つ文節を並列のキーとする．

部分並列 自立語が名詞であり，かつ名詞並列を示す助詞（表1）以外の助詞と読点を付属語に持つ文節があった場合，次の2つの条件のうちいずれかが満たされれば並列のキーとする．

1つは「その文節の助詞とそれより前の一番近い「の」以外の助詞との組が，その文節より後ろにも存在する」という条件（「〜を〜に，〜を〜に」のような場合），もう1つは「読点の次の付属語として表1に挙げた語がある」という条件（「〜を，または〜を」のような場合）である．

上記の条件で名詞並列のキーとして取り出される「〜と，」，「〜，」などは，実際には部分並列のキーとなっている可能性がある（図1の例文のような場合）．

表1 並列のキーを示す付属語
Table 1 Words indicating conjunctive structures.

名詞並列	，(読点) と も や か とか かつ だけで(は)なく および または ならびに あるいは もしくは
述語並列	の＋に対し(て) とか か し が ず＋に だけで(は)なく けれど(も) および または ならびに あるいは もしくは
部分並列	および または ならびに あるいは もしくは

注) '+' は連続する付属語であることを示す．

[*4] これにはある種の制限が課されるが，その詳細については3.4節で述べる．

[*5] 益岡・田窪文法[4]を拡張したものを標準文法とする形態素解析プログラムを使用する．この際，機能的に助詞とみなせる「という」，「に対して」，「だけでなく」などは助詞として扱う．

[*6] 益岡・田窪文法では，名詞と結合して述語を作る活用語「だ」，「である」，「です」を判定詞とよぶ．

[*7] 付属語に活用語がない場合には自立語の活用形，活用語がある場合には一番後ろの活用する付属語の活用形を，文節の活用形とする．

[*8] 益岡・田窪文法では，例えばサ変動詞について「し」を基本連用形，「して」，「したり」をタ系連用形としている．ここでいう連用形は，動詞，形容詞，形容動詞，判定詞それぞれについて，これらをすべて含めて考えている．

しかしそれらがどちらのキーであっても，同じようにその前後の文節列の類似性を調べるという方法によって並列構造を推定するので，それらが並列のキーとして区別されていないことは問題にならない．

3.3 文節間の類似度の計算

文節間の類似度は次のような5つの基準によって計算する．なお，活用する語については原形で比較する．

1. 自立語の品詞が一致している場合にポイント2を与える．以下のポイントは自立語の品詞が一致しているものについてのみ加算していく．
2. 自立語が同じ単語である場合，ポイント 10 を加える．この場合は，次の，自立語の部分一致によるポイント，分類語彙表によるポイントは与えない．
3. 自立語が名詞である場合に限り，自立語の文字列が部分的に一致する場合，一致した文字数×2 ポイント（最大 10 ポイントで打ち切る）を加える．
4. 各々の自立語について国立国語研究所の分類語彙表[5]のコードを調べ，コードの上位桁から下位桁にむかって連続する一致が3桁以上である場合，(桁の一致 −2)×2 ポイント（3の文字列部分一致のポイントと合わせて 10 ポイントで打ち切る）を加える．どちらか一方あるいは両方の自立語が分類語彙表に載っていない場合はポイントを与えることができないので0を与えている．そのような場合との間に不適切な差を生まないよう，意味的範囲が広い上位桁だけの一致に対してはポイントを与えないようにしている．
5. 同一の付属語がある場合，付属語の一致一組につき3ポイントを加える．

例えば，「低水準言語＋，」と「高水準言語＋と」では，2（品詞の一致）＋8（文字列部分一致：4文字）で 10 ポイント，また「訂正＋し＋，」と「検出＋する」では 2（品詞の一致）＋2（分類語彙表コード：3桁）＋3（付属語一致）で7ポイントとなる．

ここまでのポイントは，自立語の品詞が一致する文節間に与えられるものであるが，そうでない場合でも次のものには特別にポイントを与えた．

- 「サ変名詞＋する」などの文節は，ひとまとまりの動詞として扱っているが，その中のサ変名詞だけが「…解析，生成する …」，「…解析と 生成を行う …」のように並列構造をなしている場合がある．このような並列構造の範囲を正しく推定するためには，並列している文節（下線部分）の間に類似性を認めておく必要がある．そこで，サ変名詞を動詞として扱っている文節と，名詞として扱っている文節の間に2ポイントを与え，さらにサ変名詞同士について上の 2〜4 で計算されるポイントを与える．
- 述語並列のキーとなっている文節は，それより後ろのいずれかの文節と対応しているが，それは必ずしも自立語の品詞が同じ文節であるとは限らない（「…が 強力で，…が できる 装置を…」）．そこで，述語並列のキーの文節と，それより後ろの述語となり得る文節（自立語が用言であるか付属語に判定詞がある文節）との間に類似性を認めておく必要があるため，そのような文節間にポイント2を与える．

3.4 並列構造の範囲の推定

並列のキーの前後で，ある種の制限のもとで類似度の総和が最も大きい文節列の組を求め，それらを並列構造の範囲とする．このような文節列の組を，文節間の類似度のポイントを要素とする三角行列において以下のような方法で求める．

並列のキーの右上四角形部分を**部分行列**とよぶことにする（図1では点線内の行列に対応）．この部分行列の一番下の行の0でない要素を**起点**とし，そこから一番左の列の要素までのパスを考え，このうち後で説明する方法で計算されるスコアが最大のパス（以後これを**最適パス**とよぶ）を求める（**図2**）．この計算を部分行列の一番下の行の0以外のすべての要素を起点として行い，各起点からの最適パスのうちで最もスコアの高いパスが並列構造の範囲を示すものであるとする（**図3**）．

日本語文の各文節は意味的に右側に係っていくので，最も重要なことは並列のキーの文節が後ろのどの文節に対応しているかということである．これは，本手法ではパスの起点をどの要素にするかという問題に対応する．しかしここでは，単に「並列のキーの文節と最も類似した文節はどれであるか」というような単純な1対1の対応を考えるのでなく，各々の文節の前の部分がどのように対応しているかという広い範囲の

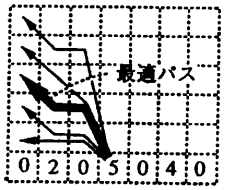


図 2 起点からの最適パス
Fig. 2 The best path from a start point.

情報を総合的に調べる．これがスコアの高いパスを求めることに対応するが，このことによって妥当な並列構造の推定を行うことができる．

以下にスコアの計算方法を具体的に説明する．

基本的スコア計算法 パスは起点となる要素から始まって各列の要素を1つずつ順につないだものである．この中で各要素をつなぐ部分を枝，枝の起点側の要素を**始点**，反対側の要素を**終点**とよぶことにする．枝としては，左横方向へのものと左上方向へのものだけを許す．あるパスのスコアは，基本的にはパス内の要素のポイントの和である．ただし，次のことを考慮する．

- 枝が左横方向である場合はその枝の終点のポイントはスコアに加えない．これは，「類似度を考慮する文節の対応としては1文節対1文節の対応だけを考える」ということを意味する．例えば，**図4**のパス '→a→b' では，文節列の対応としては 'E' と 'H I' の対応を考えているが，文節の対応としては 'a' による 'E' と 'I' の類似度のポイントだけをスコアに加え，'H' については対応するものがないと考える．

さらに次の減点を行う．

- 枝が左横方向である場合は2だけスコアを減らす．枝が左上方向で始点と終点の行の差が2以上である場合は，(行の差 −1)×2 だけスコアを減らす（図4）．これらは，「文節数が同じぐらいの，バランスの取れた並列構造がより自然である」ことを表現している．

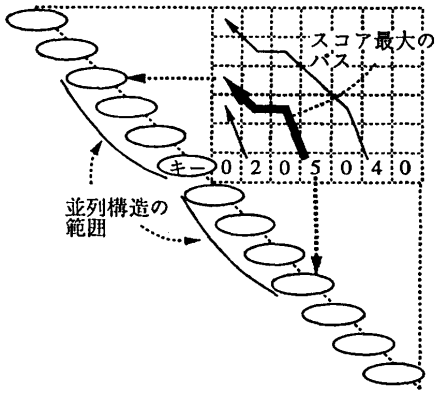


図 3 並列構造の範囲を示すパス
Fig. 3 The path specifying a conjunctive structure.

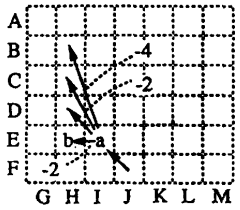


図 4 スコアの計算法
Fig. 4 Calculation of the score.

ペナルティ 2つの並列文節列の各々はその範囲内で1つの構造にまとめられることが多い．したがって，「…装置は，生産と検査の自動化を…」というような文で「〜は，」という文節が「と」による並列構造の中に含まれるとは考えにくい．すなわち「装置は，生産と」と「検査の自動化を」が並列であるとは考えられない．このような現象を考慮するために，文節列の間にある種の区切りを示す要素が存在するとスコアを減じ，それにより，さらに広い範囲にまで並列文節列がのびてゆく可能性を小さくするようにする．そのために種々の区切り要素に対して**表2**のように5段階の**区切りレベル**を設定する．そして，並列のキーの文節の区切りレベルを基準とし，それと等しいか，あるいはそれよりも強い区切りレベルの文節を並列構造に含もうとする場合にペナルティを与える．ペナルティは，(区切りレベルの差 +1)×7 とする[*9]．ただし，区切りレベルが高い場合でも，同じタイプで，しかも並列のキーとはタイプの違う2つの文節が，対応して並列構造の中に現れる場合にはペナルティを免除する．ここで，文節のタイプが同じであるとは，2つの文節において自立語の語彙以外がすべて一致すること，すなわち，自立語の品詞と活用形，すべての付属語が一

表 2 区切りレベル
Table 2 Levels of separating a sentence.

レベル	表　　現
5	「(活用語の連用形，または活用語＋接続助詞)，」「〜は，」
4	「(体言＋名詞並列を示すもの以外の助詞)，」「(副詞)，」
3	「(活用語の連用形，または活用語＋接続助詞)」「〜は」
2	「(体言＋名詞並列を示す読点以外の付属語)，」
1	「(体言)，」「(体言＋名詞並列を示す読点以外の付属語)」

[*9] ただし，レベル4の区切り要素は述語並列の範囲を限定する働きを若干持っている．そこで，述語並列（述語並列のキーの区切りレベルは5）において区切りレベル4の文節を並列構造に含もうとする場合には，特別にペナルティ4を与える．

致することを指す．このような免除を行うのは，「～は，～を行い，～は，～を行う．」のような文において対応関係にある「～は，」が並列構造の範囲を限定しているとは考えられないからである．

このペナルティは枝を伸ばしてパスのスコアを計算していく過程で次のように与える（**図5**）．

- 枝が左横方向である場合は，枝の終点の下側の文節についてペナルティを与える．この枝によって並列構造に加えられるのはこの下側の文節だけであり，またこの文節は並列構造の中で対応する文節がないので，ペナルティの免除を考える必要はない．
- 枝が左上方向である場合は，2つのことを考える必要がある．1つは，枝の終点で対応付けられる2つの文節および並列のキーの文節のタイプをチェックしてペナルティが免除されるかどうかを調べる．免除されない場合は，その2つの文節各々についてペナルティを与える．もう1つは，枝の始点と終点の行の差が2以上である場合，対応する文節なしに並列構造に含まれる枝の左側の文節についてペナルティを与える．

ボーナス 名詞並列の後の「など」のように，並列構造の直後にあって並列構造の範囲の推定に有用な表現がある．そこで**表3**のような表現が並列構造の最後の文節，すなわち起点の要素の下側の文節や，その後ろの文節にある場合，その起点からのパスのスコアにボーナス6を与える．

DP マッチング ここまで説明してきたスコアの計算方法では，ある起点からの最適パスについて**最適性の原理**が成り立っている．そこで，これをDPマッチングの手法で求める．すなわち，起点の左の列から1列ごとに計算を進め，列内の各々の要素までのスコア最大のパスを，1つ右の列の各要素までのスコア最大のパスに枝を1つ加えたものの中から求める（**図6**）．そして最終的に部分行列の一番左の列の各要素までのパスのうち，スコア最大のものを，もとの起点からの最適パスとする．

この計算を各起点について行い，各起点からの最適パスの中で最もスコアの高いパスを，並列構造を示すパスであるとする．

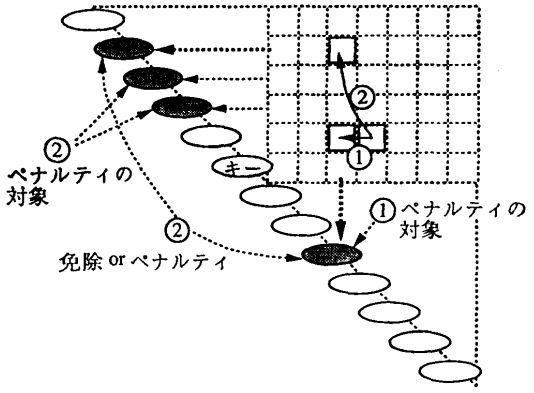


図5 ペナルティの計算法
Fig. 5 Calculation of penalties.

表3 ボーナスを与える表現
Table 3 Words for bonuses.

	名詞並列	述語並列
並列構造の最後の文節	など 等	ために ための という といった ようだ など 等
並列構造に続く文節	各～ ～種類 ～つ 組 対 両方 など 等	こと もの とき 方式 方法 手法 など 等

4. 実験結果と評価

実験は，岩波情報科学辞典，日本科学技術情報センタ（JICST）発行の抄録文，サイエンス（Vol. 17, No. 12「科学技術のためのコンピューター」），各々について，文字数が30文字以上50文字未満，50文字以上80文字未満，80文字以上，の各20文，合計180文に対して行った．

前章で挙げたように並列構造の推定には種々の要因を考慮する必要があり，各要因にどのように相対的重みを与えるかが重要である．前章で示したポイントの与え方は，情報科学辞典の例文のうち約30文に対して，正しい並列構造が推定できるように調整したものである．後に示すように，実験対象の180文全体に対しても82%の精度で並列構造を推定することができたので，現在のポイントの与え方によって各要因の相対的重みがほぼ適当に表現されていると考えられる．

4.1 解析例の説明

図7～11に解析例を示す．各例の注に示したように，連体修飾節を伴う名詞並列や述語並列では，並列

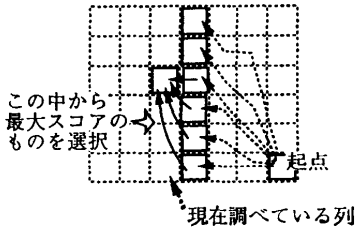


図6 DPの計算法
Fig. 6 Dynamic programming method.

構造の前の範囲がどこまでであるかについて修正を行う必要がある．ここでは，助詞「は」あるいは読点を含む文節以外のもので，並列構造内の文節に係っていると考えられる文節は並列構造に含まれるとみなしている．

表 4 解析結果の評価（並列のキー単位）
Table 4 Results of experiments by the conjunctive structure's key unit.

	名詞並列	述語並列	部分並列	計
並列のキーの数	144	118	3	265
正しい範囲を推定したもの	119 (83%)	94 (80%)	3 (100%)	216 (82%)
誤った範囲を推定したもの	25	24	0	49

図 7 の例では，1文節ごとの名詞並列が，読点同士の区切りとしてのペナルティによって正しく推定されている．3つ以上の部分の並列は，このように2つの部分の並列の組み合わせとして表現される．この例ではさらに述語並列の範囲が正しく推定されている．

図 8 の例では，3つの名詞並列が，先の例と同様に2つの並列構造の組み合わせとして正しく推定されている．

図 9 の例では，名詞並列とさらにそれを含む述語並列が正しく推定されている．この例では「計算機実験は，」と「意味では，」の「は，」の区切りとしてのペナルティ，「測れるという」の「という」によるボーナスが有効に働いている．

図 10 の例では，部分並列と名詞並列，さらにそれらを含む述語並列の構造が，**図 11** の例では，3つの文節列の述語並列の構造が正しく推定されている．

4.2 定量的評価

180文の解析結果について人手で評価を行った．180文の中に存在する並列構造に関する並列のキーは，3.2 節で示した方法によってすべて取り出された．また，並列構造でない部分から誤って取り出された並列のキーはなかった．なお，3.2 節の基準で部分並列のキーとして取り出されたものは存在しなかったが，名詞並列のキーとして取り出されたものの中に，実際には部分並列のキーであるものが3つ含まれていた（図1の文，図 10 の文がその例）．以下の評価ではこれらは部分並列として扱った．

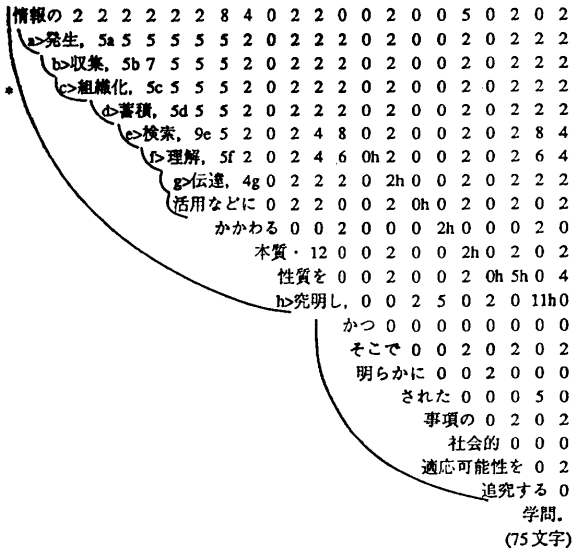
表 4 は各並列のキーに対して正しい並列構造の範囲が推定されているかどうかを調べた結果である．全体の正解率は 82% であり，この手法が十分に有効であることを示している．**表 5** は並列構造のタイプ（名詞並列，述語並列，部分並列）ごとに文単位の評価を行い，その結果を文の長さ，出典ごとにまとめたものである．1文中に同じタイプの並列構造が複数存在する場合には，そのタイプのすべての並列構造を正しく推定している場合を正解とした．この表から JICST の抄録文の解析結果があまり良くないことがわかる．これは，抄録文では限られた文字数で多くのことを記述しようとするために，人間が読んでも難解であるような文が多いためである．

4.3 失敗例と解決法

実験で正しい並列構造が推定できなかった文の具体例を示し（**表 6**），解決の見通しのあるものについてはその方法を述べる．なお表6では，下線部分が並列

表 5 解析結果の評価（文単位）
Table 5 Results of experiments by the sentence unit.

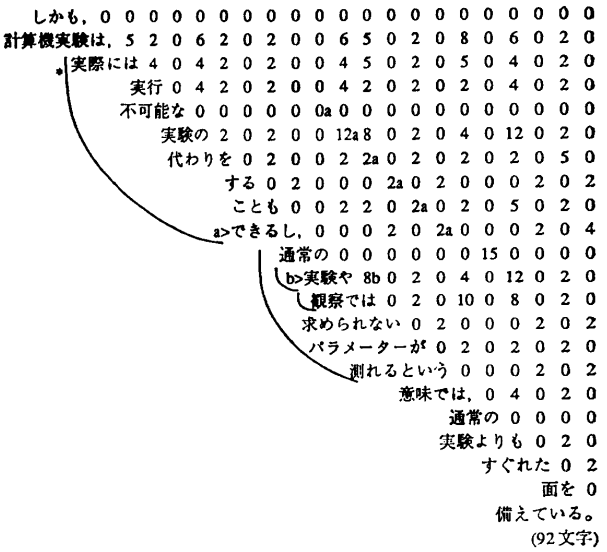
文字数 (( )内は最大文字数)	情報科学辞典			JICST 抄録文			サイエンス			計
	30-50	50-80	80-(149)	30-50	50-80	80-(144)	30-50	50-80	80-(139)	
名詞並列を含む文	8	11	11	8	11	13	5	5	11	83
正しい範囲を推定したもの	5	9	9	7	7	5	5	5	10	62(75%)
誤った範囲を推定したもの	3	2	2	1	4	8	0	0	1	21
述語並列を含む文	6	16	18	4	15	14	3	7	11	94
正しい範囲を推定したもの	6	15	16	3	10	9	1	5	6	71(76%)
誤った範囲を推定したもの	0	1	2	1	5	5	2	2	5	23
部分並列を含む文	0	2	1	0	0	0	0	0	0	3
正しい範囲を推定したもの	0	2	1	0	0	0	0	0	0	3(100%)
誤った範囲を推定したもの	0	0	0	0	0	0	0	0	0	0



*「発生，収集…」の名詞並列はひとまとまりと考え，さらにそれら全体に係る「情報の」までを述語並列の範囲にふくめる

図 7 並列構造の推定の例（1）

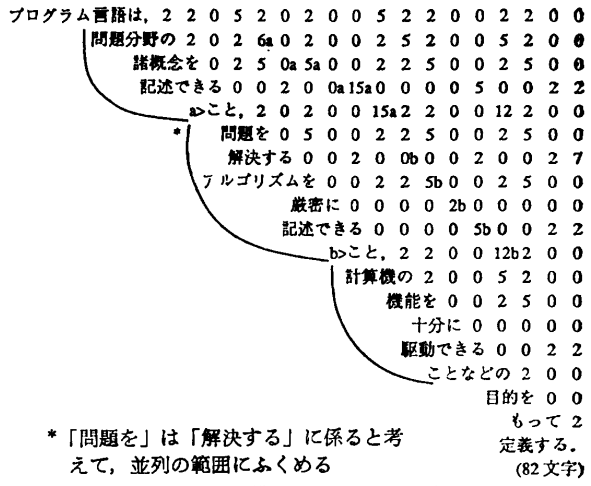
Fig. 7 An example of analyzing conjunctive structures (1).



*「実際には」，「実行」はともに「不可能な」に係ると考えて並列の範囲にふくめる

図 9 並列構造の推定の例（3）

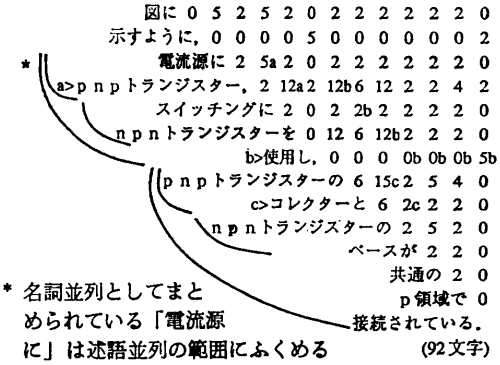
Fig. 9 An example of analyzing conjunctive structures (3).



*「問題を」は「解決する」に係ると考えて，並列の範囲にふくめる

図 8 並列構造の推定の例（2）

Fig. 8 An example of analyzing conjunctive structures (2).



* 名詞並列としてまとめられている「電流源に」は述語並列の範囲にふくめる

図 10 並列構造の推定の例（4）

Fig. 10 An example of analyzing conjunctive structures (4).

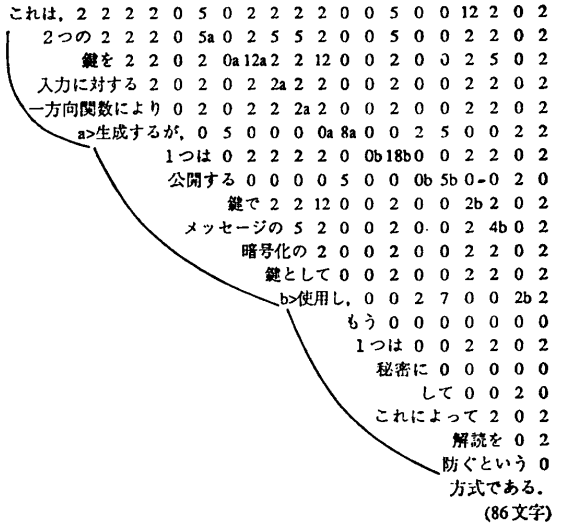


図 11 並列構造の推定の例（5）

Fig. 11 An example of analyzing conjunctive structures (5).

のキーの文節および誤って推定した範囲においてそれと並列する文節，「」が誤って推定した範囲，『』が正しい範囲である．

1. 本手法では類似した文節間に適切なポイントを与えることがまず重要である．そこで，品詞，特に名詞を数詞，固有名詞，サ変名詞，普通名詞などに細分類し，類似度の与え方に差をつけることが考えられる．例文1の誤りは，サ変名詞「拡張」と普通名詞「困難」の類似度よりも「拡張」と「保守」のサ変名詞同士の類似度を大きくするということを，他の要因とのバランスを保った上でうまく行えれば解決できる．
2. 現在，意味的類似性は分類語彙表のみを用いて与えているが，これに加えて専門用語のシソーラスなどが利用できれば1と同じ理由から並列構造推定の精度が向上すると考えられる．例文2では「アクティブ・チャート解析法」と「HPSG」の類似度により大きなポイントが与えられれば正しい構造が推定できる．
3. 推定された並列構造が統語的にみて誤っている場合がある．例文3の推定された並列構造では「文法を」の係り先がない．この例では，「文法を」と「解析と」の間に動詞がないことから，並列構造の前の範囲は「自然言語の解析と」か「解析と」しかないことが分かる（もちろん，「〜を〜と，〜を〜と」のような助詞の呼応がないことを調べる必要がある）．このような処理は，推定された並列構造の情報を用いて構文構造を決定していく次のレベルの処理として実現することを考えている．
4. 並列のキーが文のはじめであればあるほど，その後ろに並列のキーの文節と対応する可能性のある文節がたくさんあるので，並列構造の推定は難しくなる．例えば例文4のように文のはじめの連用中止が文末の述語と対応するような場合の解析は，バランスのとれた並列構造でないために非常に困難である．このような文を正しく解析しようとすれば「使って」と「製作する」の間の因果関係のような情報が必要であろう．
5. 微妙な表現であって，人間が読んでも曖昧であったり，専門的知識がなければわからないといったものがある（例文5）．
6. 解析の失敗というよりも，本手法では本質的にうまく扱えない問題として例文6のような場合がある（関係する部分を図12に示す）．このように，並列構造の前半部分に含まれ後半部分には対応するものがないような文節列がある場合，正しい並列構造を推定することができない．

例文5や例文6を正しく解析しようと思えば深い意味解析を行う以外に方法はないが，それは言語情報処理の最終目標の1つであろう．

なお，実験した例文については失敗例がなかったが，「〜，さらには」「〜，例えば」のように挿入句的な形で存在する並列構造は，本手法ではうまく扱えない場合がある．この取り扱いについては別に考慮中である．

表 6 解析の失敗例
Table 6 Examples of failure of analysis.

例文 1：これら解析手法の共通した問題として文法規則が大きくなった場合の「規則の『拡張や 保守の』困難が」上げられる．(49文字)
例文 2：ATR 自動翻訳電話研究所で開発された音声言語日英翻訳実験システム SL-TRANS の日本語対話文解析部は，『「解析過程の制御が自由なアクティブ・チャート解析法と 単一化に基づいた 語い-統辞的な文法的 枠組みである」HPSG を』採用している．(113文字)
例文 3：「単一の文法を自然言語の『解析と 生成に』用いる双方向文法の 研究は，」計算言語学の上からも，機械翻訳や自然言語インタフェースといった応用面からも重要である．(73文字)
例文 4：実際，筆者たちは『「これを 使って，重力相互作用が 支配する」天体の運動について，高精度で高速の数値計算ができるディジタル・オレリーという専用コンピューターを製作している．』(81文字)
例文 5：一般に，生成アルゴリズムが完全であることは証明できるが，『「非文に対する 停止性や 出力する文の あいまいさの」上限について』保証がない．(62文字)
例文 6：述語慣用句については，『表現ごとに用意した「結合価構造中の 結合価要素と 文中の 格要素」』との対応を取ることにより解析し，機能動詞表現については，意味の大部分を動作名詞が担っている点に着目し，動作名詞の用言形を述語とする文として解析を行う．(113文字)

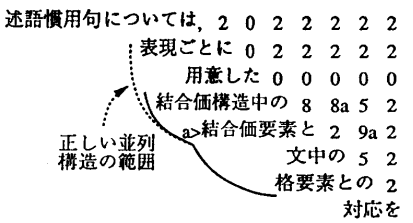


図 12 並列構造推定の失敗例
Fig. 12 An example of failure of analysis.

5. おわりに

以上に示したように，長い文の内部構造として多く存在する並列文節列が，文節列同士の類似性の発見という考え方でかなりうまく検出できることが分かった．この方法の導入によって多くの長い日本語文が正しく解析されるようになるだろうが，それでも解析に失敗する場合はまだまだ残る．しかしそのような失敗についても，深い意味解析に行かずに表層的な手係りによって解決できる場合はいろいろあると考えている．単純な文法規則しか導入せず，それで多数の解析結果が出たり，解析に失敗したらすぐ意味処理によって解決しようとするのは，いささか安易な考え方であると思う．意味素性を導入し，意味的整合性をチェックするという方法もよほど精密なものを作らない限り実際にはそれほど効果を発揮しない[1]．他に有効な意味処理の方法は現在のところないのであるから，できるだけ構文的な現象を綿密にしらべることが必要である．これをひと口でいえば，1つの文，あるいは複数の文の並びにおいて，できるだけ広い範囲の言語表現を同時的に調べるということであろう．本論文ではその1つの手法を示し，それが有効であることを示した．

参考文献

1) 池原，宮崎，横尾：日英機械翻訳のための意味解析辞書，情報処理学会自然言語処理研究会報告，84-13 (1991).
2) 長尾，辻井，田中，石川：科学技術論文における並列句とその解析，情報処理学会自然言語処理研究会報告，36-4 (1983).
3) 首藤，吉村，津田：日本語技術文における並列構造，情報処理学会論文誌，Vol. 27, No. 2, pp. 183-190 (1986).
4) 益岡，田窪：基礎日本語文法，くろしお出版 (1989).
5) 国立国語研究所：分類語彙表，秀英出版 (1964).

(平成3年9月30日受付)
(平成4年6月12日採録)

黒橋 禎夫（正会員）

1989年京都大学工学部電気工学第二学科卒業．現在，同大学院工学研究科博士後期課程在学中．自然言語処理，知識情報処理の研究に従事．人工知能学会会員．

長尾 眞（正会員）

1960年京都大学工学部電子工学科卒業．1962年同大学院修士課程修了．京都大学工学部助手，助教授を経て，1973年より同教授，現在に至る．1976年より国立民族学博物館併任教授．パターン認識，画像処理，自然言語処理，機械翻訳等の研究に従事．