ＬＴＢにおける構文解析システムＳＡＸについて

山崎　重一郎、　　弘田　直人、　　赤坂　宏二
富士通　　日本情報科学研究所　ＩＣＯＴ

1．はじめに

ＩＣＯＴでは、談話処理システム研究のベースとなるソフトウェアツール群を汎用日本語処理系（ＬＴＢ）として研究開発を進めている。ＳＡＸは、ＬＴＢの構文解析ツールであり、ＤＣＧによる文法をもとに上昇型の解析を行う並列構文解析システムＡＸの逐次版システムである。本稿では、ＬＴＢにおける、ＳＡＸと形態素解析システムＬＡＸや意味処理言語ＣＩＬとの連携方法と、ＳＡＸの実行及びデバッグの環境について述べる。

2．ＬＡＸとの連携

ＬＡＸは並列形態素解析システムの逐次実行版システムである。ＬＴＢは利用者とのインターフェースとしてＬＴＢシェルを備えており、ＬＴＢの各ツールはＬＴＢシェルのパイプを通じてデータを交換できる。ＬＡＸによる形態素解析の結果は、ＬＴＢシェルのパイプを通じてＳＡＸに送られる。このＬＡＸからの入力は、形態素解析の結果得られた統語範疇の並びであり、ＳＡＸによる構文解析はこれらの統語範疇の並びより上の構造の解析を行う。ＬＡＸは、形態素解析に曖昧性がある場合、その全ての解を束状に縮退した統語範疇の並びを結果として与えるが、ＳＡＸはこの形式の入力をもとに解析を実行することができる。

ＬＡＸの解析結果の例

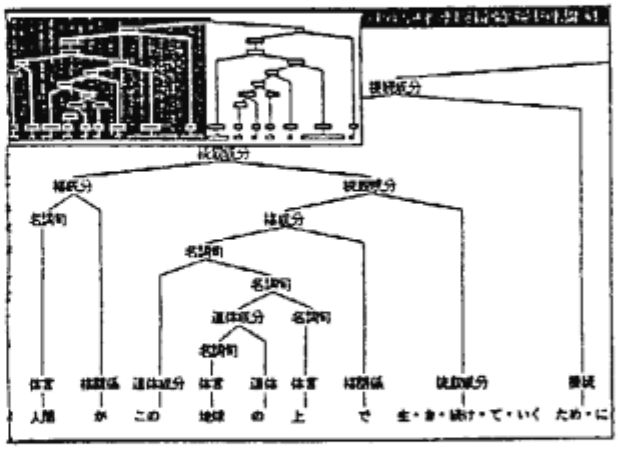
```
人間がこの地球の上で生きつづけていくためには、...
[体言（｛語彙／人間｝），
 格関係（｛語彙／が｝），
 連体（｛語彙／この｝），
    ・・・
（統叙成分（（語彙／［生き，続け，て，い，く］...），
  ［統叙成分（（語彙／［生き］｝....），
    統叙成分（（語彙／［続け，て，い，く］｝... ），
 接続（｛語彙／ため｝），
  ・・・］．
```

3．ＣＩＬとの連携

ＣＩＬは、部分項というデータ構造に対する単一化と遅延評価による制約記述によってＰＲＯＬＯＧを拡張した論理型言語である。ＣＩＬの部分項は、ラベル／データ対を節とし横方向に順序を持たない無限木構造に対応するデータ構造であり、部分項による記述方法や部分項の単一化は自然言語の意味処理に適している。ＳＡＸは、ＤＣＧの補強項としてＣＩＬのゴール列を記述することや、ＤＣＧ規則の中にＣＩＬのプログラム節を記述することを許しており、これらの補強項で意味処理を行うことよってＳＡＸによる構文解析と同時に平行して意味処理を行うことが可能になる。また、ＳＡＸの解析アルゴリズムは決定的であるので、意味処理に非決定的な部分がある場合多重環境が必要になるが、遅延実行の利用により環境の複写を最小限にすることができる。これらのＣＩＬプログラムの評価は、ＳＡＸの解析実行部が、補強項の評価が必要となった時点で、ＣＩＬインタプリタを呼び出すことによって実現している。

4．解析実行環境

ＳＡＸの解析実行環境は、ＬＡＸからの入力をもとに全ての可能な構文木を求め、得られた木構造の数を表示し、それらの木の縮小表示をメニューとして表示する。利用者はこの縮小表示をマウスで選択することによって詳細な木構造をウィンドーに表示させることができる。また解析結果の木構造が巨大なためにウィンドーに一度に表示しきれない場合は、縮小表示によって表示範囲を指定することも可能である。また、解析の結果得られた意味構造は、部分項のプリティプリントとして表示させることができる。



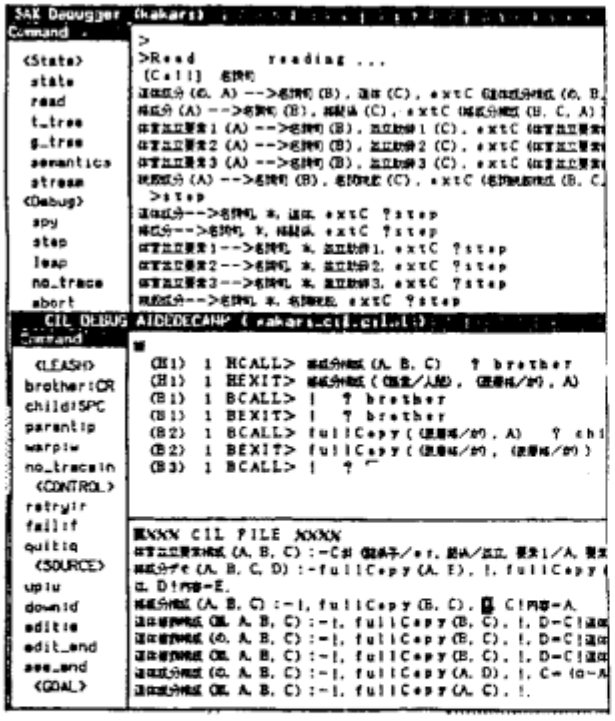
The parsing system SAX in LTB
Shigeichiro YAMASAKI[1] ,Naoto HIROTA[2] ,Kouji AKASAKA[3]
(1)Fujitsu Ltd. (2)Information Science Labs. (3)ICOT

5．文法デバッグ環境

ＳＡＸの文法デバッグ環境は、実行トレースによる動的デバッグ環境と解析実行後に解の探索履歴を用いて解析過程で構成された木構造を再構成する静的デバッグ環境の二つの環境を備えている。この二つのデバッグ環境は、互いに補完的役割を担っている。静的デバッグ環境は、解析の失敗位置を特定するために使用される。これは、ＳＡＸの解析が同時に多数の木を成長させるために、トレースによって解析の失敗位置を発見することが困難であるからである。一方、動的デバッグ環境は、補強項の評価の失敗や、遅延実行による影響を発見するために使用される。

5．1　動的デバッグ環境

ＳＡＸの動的デバッグ環境におけるトレースは、ＳＡＸの実行過程を、ＤＣＧ規則にその時点での解析位置を表す'＊'記号を埋め込んだ式の系列として表示される。動的デバッグ環境では、メニューからの選択により、文法に現れる任意の統語範疇およびＤＣＧ規則にスパイポイントを設定することができる。これによって、スパイポイントが設定された統語範疇や文法規則までトレースをリープすることができる。補強項のＣＩＬプログラムの評価を行う時は、ＣＩＬインタプリタのデバッガに制御が移り、ＣＩＬプログラムのトレースが行われる。



5．2　静的デバッグ環境

静的デバッグ環境は、解析実行後に、利用者が、その解析で構成に失敗した構文木の部分構造を、解析に使用された文法規則のメニューの選択によってボトムアップに再構成していき、そのメニューに期待している文法規則が出現しなくなった位置を知ることによって解析の失敗位置を特定する環境と、利用者による入力列への範囲指定により、その範囲から実際に構成された部分木の構造を表示する環境から成っている。メニュー選択による構文木の再構成の開始位置は、入力列の間の位置を選択することによって決定される。メニュー選択によって再構成されるのは、この開始位置の左側の要素の上位に構成された不完全な木構造のみなので、失敗位置を特定するために、利用者は、何度か開始位置を変更する必要がある。したがって、利用者は、まず範囲指定により心に描いている木構造の部分構造の構成の成否を検査し、失敗位置の大まかな範囲を知った上で、メニュー選択による構文木の再構成を行うことによって，効率的に失敗位置を特定することができる。

6．おわりに

ＬＴＢにおけるＳＡＸの環境について述べた。今後、実際の文法開発を通した改良と、デバッグ機能の拡張を行っていき、快適な文法開発環境と効率の良い解析実行環境を実現し、文脈処理などのより高度な自然言語処理の基盤となるツールとして、多くの方に使用してもらえるものにしたい。

謝辞

本研究は第５世代コンピュータプロジェクトの一環として行われ、ＩＣＯＴ第２研究室の内田氏、吉岡氏、杉村氏をはじめとする方々に御支援を頂きました。ここに印して感謝いたします。

【参考文献】

〔１〕松本裕治，杉村領一：論理型言語に基づく構文解析システムＳＡＸ，ソフトウェア科学会論文誌 1886, Vol.3 No.4

〔２〕向井国昭，奥西稔幸，天沼敏幸，鈴木隆之：ＣＩＬ言語マニュアル第２版，1987

〔３〕杉村領一，赤坂宏二，久保幸弘，松本裕治，佐野洋：論理型形態素解析ＬＡＸ，LPC 1988

〔４〕山崎重一郎，杉村領一，赤坂宏二，松本裕治：構文解析システムＳＡＸのデバッグ環境，ICOT TR