

言語処理系教育支援ツール

林 恒 俊†

†立命館大学情報理工学部

1 はじめに

情報関係学科において言語処理系の学習は基礎的素養と考えられている。プログラミングは重要な技術である。作成された原プログラムがどのような段階を経て実行されるかを理解することはプログラミングを学習する上で欠くことができない。さらに言語処理系を構成する様々なアルゴリズムは高度なプログラミング技術の基礎になるものである。

しかしこれを始めて受講する情報系の大学生にとってもかなりハードルの高い講義のように見受けられる。原因は明白で高校までのカリキュラムでは言語処理系の基礎を構成している理論について一切触れられないからである。

例えば電気工学なら高校卒業までに代数幾何解析の基礎を学習しているため電気回路理論の導入は容易と考えられる。しかし記号列、言語、オートマトン、文法等の基礎がない状態で言語処理系を理解するためにはかなりの量の学習を必要と推測される。

また国内では学習後実験や演習で実習する機会があっても本格的に学習したりあるいは卒業研究の対象にすることはほとんど考えられない。これは国内での言語系の開発が諸外国に比べて非常に乏しい原因・結果とも考えられる。

これらの点から言語処理系学習者の理解を援助するため言語処理系の動作を容易に再現するツールを実装することを思いついた。以下でこのツールについて報告する。

2 支援ツール導入経緯

言語処理系の理解を援助するツールを導入した理由付けは以下のようである。

もともと言語処理系は多様なアルゴリズム上に組立てられているため言語処理系を理解するためにはアルゴリズムと同時にそれを実際の課題に適用できなければならない。

このようなアルゴリズムの代表的な例は構文解析の様々な技法である。構文解析技法自身はアルゴリズム

を文章で説明することが可能である。実際に原プログラムに適用して具体的に構文解析を実行するためには適切な演習が必要と思われる。

例えば次の LL(1) 構文解析表により与えられた式の構文解析技法を学習する場合

LL(1) 構文解析表

	a	$+$	$*$	$($	$)$	$\$$
E	TE'			TE'		
E'		$+TE'$			ϵ	ϵ
T	PT'			PT'		
T'		ϵ	$*PT'$		ϵ	ϵ
P	a			(E)		

LL(1) 解析法を説明し次のような解析例 [2] を示すことが可能である。

LL(1) 解析実例

スタック	入力記号列	注 釈
E	$a + a * a \$$	初期状態
TE'	$a + a * a \$$	$T \rightarrow PT'$
$PT'E'$	$a + a * a \$$	$P \rightarrow a$
$aT'E'$	$a + a * a \$$	一致
$T'E'$	$+ a * a \$$	$T' \rightarrow \epsilon$
E'	$+ a * a \$$	$E' \rightarrow +TE'$
$+TE'$	$+ a * a \$$	一致
TE'	$a * a \$$	$T \rightarrow PT'$
$PT'E'$	$a * a \$$	$P \rightarrow a$
$aT'E'$	$a * a \$$	一致
$T'E'$	$* a \$$	$T' \rightarrow *PT'$
$*PT'E'$	$* a \$$	一致
$PT'E'$	$a \$$	$P \rightarrow a$
$aT'E'$	$a \$$	一致
$T'E'$	$\$$	$T' \rightarrow \epsilon$
E'	$\$$	$E' \rightarrow \epsilon$
空	$\$$	受理

しかし例に挙げた入力以外のデータについて学習者に解析を実行させてみるとかなり心もとない。できるだけ多数の例を使用して説明を行うことが望ましいが時間的な制約等もあり難しい。

学習者が主体的に様々な解析例を試してみることができればこのような解析アルゴリズムの理解に有用であ

Supporting Tools for Teaching Language Processors

†Tsunetoshi HAYASHI

†Faculty of Computer Science and Engineering, Ritsumeikan University

ると考えられるので今回の支援ツールを実装した。ツールの有効性についてはこれから検討する予定である。

3 ツール実装方式

支援ツールは次のような機能が実現されていることが望ましい。

- ツールは講義時間とは独立に常時に利用できることよい。できれば講義時間以外に利用できることが重要である。
- ツールは場所を限定せずに利用できることよい。できれば自宅で利用できれば予習復習にも役に立つはずである。
- ツールはできるだけ柔軟なプラットフォーム上で動作することが望ましい。実際にはパーソナルコンピュータ上での利用ということになる。

ここでは次のような支援ツールの実装が考えられる。

- プラットフォーム固有アプリケーション
- Java アプリケーション
- Web アプリケーション

上記の条件を満たすためには特定プラットフォーム上のアプリケーションとしてツールを実装することは適切ではない。

Java アプリケーションはプラットフォーム依存性が少ない。しかも最終的に学習者のパーソナルコンピュータにインストールしなければならない。この点はかなりのハードルになる。

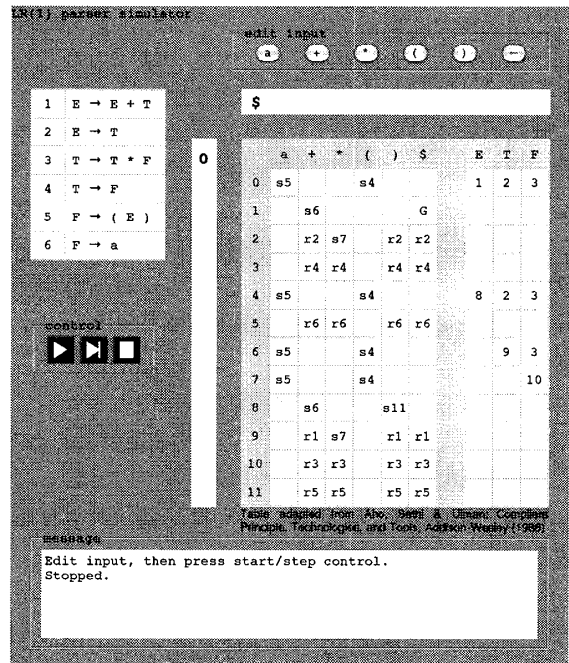
もし要求される機能を正常に実装可能なら Web アプリケーションによる実現が最も望ましいものと思われる。支援ツールは次のような機能に対する要求を満たさなければならない。

- 適切なデータを入力・編集可能である。
- アルゴリズムを代表する要素を明確に表示する。
- 解析を段階を追ってあるいは連続して実行可能である。
- 解析実行中の状態を常に表示する。
- 査収結果が表示される。

次に示す例はこれらの条件を満たすように実装された LR(1) 解析系 [1] の Web アプリケーションの例である。実際にブラウザで表示した画面を図に示している。

初期状態では LR(1) 解析表、番号付き生成規則、入力領域、スタック領域等が示されている。また入力を

編集するためのボタンが用意されている。さらに解析をステップ実行、連続実行、停止するための制御ボタンも供えている。



LR(1) 解析ツール

ツールの実体は主 HTML ファイルといくつかの CSS 及び JavaScript ファイルから構成されている。次のブラウザ Firefox、Safari、Opera で動作確認を行った。

4 評価とまとめ

現在まで完成したツールは LR(1) 解析法のみである。今後に次のようなツールを作成する計画である。

- LL(1) 解析ツール
- 帰納下降型解析ツール
- 演算子順位上昇型解析ツール

さらに有限状態機械に関するいくつかのアルゴリズムや逆ポーランド記法評価についても類似ツールを作成することを考えている。

LR(1) 解析ツールのみが完成していて実際の講義で十分にテストすることはできなかったが全ツールが完成すればかなり有効だと思われる。

参考文献

- [1] A V Aho, R Sethi & J D Ullman, *Compilers : Principles, Techniques, and Tools*, Addison-Wesley, 1986.
- [2] 黒住 及び 林, システムソフトウェア, オーム社, 2001.