

TeX 数式を含んだ文章のEbra を用いた点訳

宮川正弘, 染田貞道*, 半田剣一**

(視覚部情報処理学科, *元視覚部情報処理学科, **電子技術総合研究所)

要旨: 点字エディタ Ebra は TeX の数式表現を直接数学点字に直すことができる。TeX 数式を含んだ文章を Ebra で処理するためのプリプロセッサ t2e を作成した。これにより数式を含んだ原稿の点訳が正確になり、要する時間も大幅に短縮できた。

1. はじめに

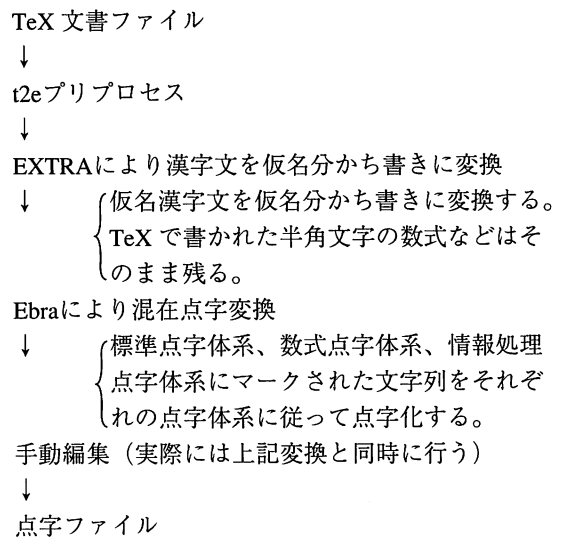
授業には拡大文字の教材と点字の教材をそれぞれ用意する。拡大文字の教材は TeX システム を使って作成する。数式は Σ などのように特殊な記号や書き方を含み、これらに対応する点字は数学点字として定められている[4]。TeX (Knuth 1982) では数式 Σ_i は文字列として `\sum_i` と表記される。混在点字エディタ Ebra [1] は点字の対話的編集を主目的としたが、その変換機能を使って TeX の数式表記を直接対応する数学点字に直すことが出来る。しかし Ebra に使用すべき点字体系を指示するためには文中に対応する制御文字を挿入しなければならない。そこで、この作業を自動化する目的で TeX ファイルに Ebra 用の制御文字を自動的に挿入するプリプロセッサ t2e (tex-to-ebra の意) を Emacs lisp を用いて作成した。プリプロセッサは数式の開始と終わりをマークするだけでなく、正しくて読みやすい点字印刷が出来るように、TeX の制御文字列を削除・変換したり、数式内に処理を加えて EXTRA[3] の自動点字清書機能を補助している。

プリプロセッサは TeX (LaTeX) の全ての機能をカバーして、点字出力を全自動化する試みではない。ここでは LaTeX の自作のマクロや自分が使用するマクロだけに対応した処理しか実装していないが、日常授業用の点字原稿の作成は、最終的な編集だけを行えばよく、LaTeX の原稿を手動で編集していた時に比べ手間は大幅に減少した。

2. 点訳処理の流れ

処理の流れを図1に示す。漢字仮名混じりの文を仮名分かち書きにする処理は EXTRA の変換機能を使っている。数式や制御文字列には半角文字が使われているが、EXTRA はこれらを変換せず、全てそのまま残す。そのため、改行(半角文字)もそのまま保存される。TeX 文章では改行は自由に使ってよい。したがって、TeX 文章の中で自由に使われた改行の中で余分な改行を削除した

図1 Tex文書を点訳する処理の流れ



文章を EXTRA に渡すようにしなければならない (3節で詳述)。

3. t2e の処理の実際

表2に主な処理と対応する自作のマクロコマンド、あるいは LaTeX コマンドのリストを示す。t2e は処理すべきコマンド群に属するコマンドが入力ファイルの中に出現しているかどうかを、ファイルの先頭から調べていく。一つの類のコマンド群は1つの正規集合として表わされ、Emacs Lisp の re-search-forward (正規集合検索命令) が使われている (文献[1]を参考にした)。TeX はユーザに極めて自由な形の文書作成を許す。例えば、TeX 文書では改行文字は無視されるので、ユーザは自分の見やすいように自由に改行した文書を作成する。一方 t2e の出力を処理する EXTRA は全ての改行文字を有効とするので、t2e は必要な改行文字のみを残すようにしなければならない。必要な改行とそうでない改行の認識は、具体的には、隣り合った2つの空行で挟まれた文章群を

paragraphとして認識し、1つのparagraphを1行に出力する事により達成した。t2eのこの他の具体的処理を表2の関係する項目番号と共に以下に列挙する。

(4) TeXの制御文字の除去（項目4と5に示すコマンドの中の\largeなどのように両方の形で使われるものは、区別して処理する）、(6,7,8) itemize, enumerate, descriptionなどの字下げ、(11) 式の番号付けと引用する式番号の半自動処理、(14) 空行挿入削除の制御、(15,16,18)数式部分の開始と終わりにEbraの制御文字を挿入して明示する。(15,16)特に改行した数式(display math)については前後に空行を挿入している。(17)入れ子した記述を許す分数形式 $\frac{A}{B}$ などを表す文字列 $\frac{A}{B}$ は $A(x/y)/B$ の形に変換される（この機能はEbraにも実装済みであるので使わなくても良い）。

表3にt2eの入力例（TeXの文）をまた表4に対応する出力例を示す。この出力がEXTRAへの入力となり、EXTRAにより仮名分かち書きに変換される。

4. プリプロセス t2e の特徴

最大の特徴は強力な文字列処理システムである Emacs Lisp 言語で書かれていることである。これにより次の利点を持つ。

- (1)普及したフリーソフトエディタ Mule (emacs) が使えればコンピュータの機種を問わない汎用性がある。
- (2)入れ子構造の処理が簡単にできる。

5. おわりに

点字用紙は一行に32文字しか収容できない。通常の表をこのような制限の下で点字の形式で判りやすい表に点訳する事は、点訳者が工夫しているところである。このような問題解決が点字自動タイプセットの課題である。

参考文献

- [1] B. Martenson: DeTeX, UNIXフリーソフト, 1987.
- [2] 染田貞道: 「情報処理・数学記号および英語混在文章用点字エディタ」, 筑波技術短期大学テクノレポート4, 103-105, March 1977.
- [3] 石川 准: 「EXTRA 使用説明書(v.1.5)」, アメディア, 1991.
- [4] 日本点字委員会 (編): 「点字数学記号解説」, 日本点字委員会発行, 1981.

表2

	処理	LaTeXコマンド
1	コメント行除去	%で始まる行 (\%を除く)
2	右寄せ	rflush
3	コマンドとそれに続く開括弧 {と対応する閉括弧} を除去	kougi. theme. sub-theme. lflush. mbox. framebox. parbox
4	文字の大きさなどを指示する行コマンドを削除	small. large. medskip. bigskip. normalsize. earr
5	スコープを括弧 {と} で表した宣言コマンドを括弧ごと除去	em. bf. bm. gt. sl. Large. large
6	字下げと自動番号付け	enumerate
7	字下げ	description
8	字下げと特殊文字挿入	itemize
9	引数ごと除去	barr
10	空白に変換	& (表中)
11	式番号	labeq. refeq
12	改行に直す	\\
13	横線 (……) に直す	hline
14	1パラグラフを1行に	EXTRAの要請
15	数学環境の開始	\\[. beq. beqar. beqarst
16	数学環境の終了	\\]. eqq. eeqar. eeqarst
17	分数式	frac
18	数式の始まりと終了	\$

表3 t2e の入力

```
\rflush{12.5,1998}
\theme{レポート 1}
\medskip
\beq \sum_{i=1}^n |x_i-y_i|\labeq{1} \eeq
関数  $\text{d}()$  は性質\refeq{1}を満たす.
```

表4 t2e の出力

```
12.5,1998
レポート 1

 $\sum_{i=1}^n |x_i-y_i|$  (式 1)
関数  $\text{d}()$  は性質 (式 1) を満たす.
```