

# 2B-8 ALPS-REDUCEの インプリメンテーション

井田昌之・大浦範之・小林茂男・遠峰隆博  
(青山学院大学 理工学部 経営工学科)

## 1. 概要

数式処理言語REDUCE<sup>[1]</sup>は、ユタ大学のA.C.ハーンにより作成されたもので、LISPをホスト言語とした専用言語である。日本でも東大計算センターを始め、いくつかの計算機にHLISPをホスト言語としておこなわれている。REDUCE言語は、その処理系自体が数十Kセルを要するので、大型機でなければ使用できなかった。

筆者等は、かねてからマイクロプロセッサを使用したLISP専用機ALPS/I<sup>[2]</sup>を開発してきた。ALPS/Iは学内におけるLISP処理系の提供を目的としていた。(本学の計算センターの大型機にLISPをのせ、使用することは運営上好ましくなかったのも一因である。)そして、数式処理言語は以前より求められており、REDUCEをALPS/Iにのせることは将来的見地からも重要と考えられた。その第一歩として、ALPS-REDUCE初版をのせ、作動させることに成功したので、ここに報告する。ALPS-REDUCEの移植は、システム作成者の立場からすれば、ALPS/Iの実用性の1つの目安として(メモリ容量に関して)非常に意味がある。その意味からALPS-REDUCE初版の移植は成功したといえ、現在REDUCE-IIの全機能を含めるように予定したALPS-REDUCE第2版の作成を進めている。(本年度中に完成予定。)

## 2. REDUCEの移植過程 について

ALPS/IにのせられたREDUCEは、東大 寺島氏(現在電通大勤務)により頂いたソースプログラムを基にしている。

次の手順により行なわれた。

①ソース中の行列演算関係の部分の削除

REDUCEのサイズは、約50Kセルと想定されたので、移植に先立って極力プログラムの縮小が必要と思われ、その部分の切り落としをした。結果的には、これをしたために、手続き関係REDUCEのProcedure)が出来なくなった。また、現在の所要セルの総

表-1 ALPS/Iにのせた関数とその対応系

function名	対応系
*APPLY	APPLYのSynonymとして定義
*EVAL	EVALのSynonymとして定義
DELETE	DEFINEで定義
DISGRT	DEFINEで定義
DISVDE	QUOTIENTを用いてDEFINEで定義
EXPT	MULTを用いてDEFINEで定義
FLAG	flag[x;ind] = mapc[x; λ[[i]; seta[ind; i; T]]]
FLAGP	flagp[x;ind] = errset[ind[x]; T]
GET	get[x; y] = y[x]
PUT	put[x; y; z] = seta[y; x; z]
QTS	qts[x] = and[eq[get[x]; APVAL]; gval[x]]
PTS	pts[x; y] = cset[x; y]
GETD	getd[x] = gval[x]
LATER	ORDERPを用いてDEFINEで定義
NEG	機械語にて組み込み
ORDERP	機械語にて組み込み
PLUS	SUMのSynonymとして定義
PRINTTY	printty[x] = and[not; print[x]]
RECIP	QUOTIENTを用いてDEFINEで定義
REMAINDER	QUOTIENTを用いてDEFINEで定義
REMFLAG	remflag[x; y] = mapc[x; λ[[z]; delete[y; z]]]
REMPROP	remprop[x; y] remflag[x; y]と同じ
TIMES	MULTのSynonymとして定義

\*: Synonym 処理というのは、そのAtomの値部及び属性部をすべて定義し、同じものと同一にするということ

[例] ATOMHEADをCARのSynonymとすれば、次の2つの関数を実行する。

```
pair[atomhead]; get[car]      : 属性部のコピー
eval[atomhead]; gval[car]    : 値部のコピー
```

この結果ATOMHEADは余分なoverheadなしにCARと同じに使用できる。

教は35Kセル(H-area等を含めれば43K語)であるが、切り落とし部分を追加しても教Kセルで済むので、この切り落としは不用であった。

②ユーティリティの作成

エディター及びプリティプリンタを作成し、ソースプログラムの解説・書き換えに役立った。

③幾つかの関数の作成 (表-1参照)

④フラグに用いられる識別子及びPropertyの機能のハッシュ化配列による代わりを(FLAG, FLAGP, GET, PUT関数の第2引数に用いられているもの)の名前を洗い出して、宣言文を作成した。

⑤デストロイディング

以上の加工処理を受けたソーステープを、ALPS/Iに読ませた。この結果アトムストリングエリアが不足したので、その領域の増加を行ない、Systemの再編集をした。

⑥動作を確認し、処理系をカセットMT上にのせた。

3. ALPS-REDUCE初版の動作状況

ALPS-REDUCE初版では、数式の展開・整理及び微分が可能である。SystemのLoadingは約17分を要する。以下に実行時のデータ及びリストを示す。

- ① 所要メモリ総数は約43K語(リスト領域35K語、うちシステムは約30K語)、自由領域約21Kセル。
- ② Loading時のH-分子使用量は1317個(うちProperty及びフラグ名は52個)。
- ③ DEFINEした関数の数394個。
- ④ PNAMEのoverflow領域の使用量1070語。

(A-B-C-D)\*\*3;                      図-1 (式の展開の例)

$$A^3 - 3A^2B + 3A^2C - 3A^2D + 3A^2B^2 + 6A^2BC + 6A^2BD + 3A^2C^2 + 6A^2CD + 3A^2D^2 - 3A^2B^3 - 3A^2C^3 - 3A^2D^3 + 6A^2BCD - 3A^2B^2C - 3A^2B^2D - 3A^2C^2D - 3A^2D^2C - 3A^2D^2D$$

実行時間(図-1) 結果のプリント開始まで 33秒  
 プリント終了まで +25秒  
 使用したH-area 172個  
 カバレッジコレクションは多数回働く

DF((2\*X\*\*2+3\*X-4)\*\*3,X);                      図-2 (微分の例)

$$3*(16*X^5 + 60*X^4 + 8*X^3 - 117*X^2 - 8*X + 48)$$

また、容量的にみるために(X+Y+Z)<sup>10</sup>の展開を行なった。(応答時間で比較) 理研のHLISP-REDUCEで約15秒(TSSオーバーヘッドを含む)でプリント開始。ALPS-REDUCEでは300秒でプリント開始(終了までにさらに370秒)。H-分子使用量は176個。(この程度の大きさは十分可能である。なお、プリント時にPRINT文の小さな"虫"が見つかった)

謝辞: 日ごろ御指導いただきまます経営工学科間野浩太郎教授並びに日夜ともに戦いつづけた森芳喜, 善本正一君をはじめ同研究室の諸兄に深謝いたします。

参考文献 [1] A.C.Hearn REDUCE-II USER'S MANUAL 1973 Utah Univ.