数理解析研究所講究録
第612巻 1987年 94-100

数式処理に於ける ヒューマンフレンドリネス

大阪電気通信大学工学部　　対馬勝英（Katsuhide Tsushima）

I．序

数理研究のツールとしてヒューマンフレンドリな形で数式処理システムを利用するには、

1)紙の上でのアルゴリスム，記号法，表記法がそのままの形で数式処理システム上で使える。

2)柔軟な作業環境が用意されていること。

3)利用，処理結果の評価,，必要な数学知識に関する助言をシステムより入手出来ることが必要である。

1)は数式処理システムの本質に関係することであり、Ⅲ章にて詳しく述べる。2)はI/Oの改善とエディタ，ヒストリ表示機能，シエル機能など数式処理システムの持つべきユーーティリティに関連したものが多い。これについてはⅡ章において論じるが、それ以前に数式処理システムの利用者，利用目的について述べ、数式処理システムに対する要求を分析しておく必要がある。（Ⅰ章）

I．ユーザモデル

ユーザのレベル，目的により数式処理システムの利用に関して大きな差異が生じる。1表にその分類を示し、それに応じてヒューマンフレンドリネスの概念が異なることを論じたい。

	用途	処理速度	機能	数学的説明機能	システム処理説明機能	ユーザによるシステム変更可能性
A	数学教育	遅くてよい	汎用	必要	不要	不要
B	学習者用	普通	汎用	必要	必要	あってもよい
C	専門家用	速い	汎用	殆ど不要	必要	必要
D	特殊用途	最も速い	特殊	不要	必要	必要

また，利用形態も大別して，

1)ノート替わりに殆どの紙上の処理をコンピュータとの対話に移行さす。

2)特殊な特定の長大な処理ができればよい。

3)数学的な知識の検索，または確認

があげられる。 (2表)

この3 つの利用形態の差により、数式処理システムに要求される機能とフレンドリネスの内容も変化する。従来の多くの数式処理システムは、整式の処理や整式ベースの発想が多く、2)の目的を持つ高レベルのユーザを想定して行われてきた。これはコンピュータのマンマシンインタフェースが貧弱であり、かつコンピュータ端末を長時間(終日)，個人的に専有できなかった時代背景を反映している。

パソコン上の小型システムの利用形態は1),3) に重点が置かれているし、REDUCE等の大型システムも個人の専有の出来るパソコン上のシステムの場合には、2)から1),3) を含んだ形の応用を含んだ形への利用形態の移行が目立つ。

利用形態	専有度	利用者レベル	数学知識DB	柔軟なI/O
1)	四六時中	広範	必要	最も必要
2)	スポット的	専門家	不要	不要
3)	四六時中	大学生以上	必要	必要

第2表　　利用形態と必要な機能

1 表と2 表を併せて考えると、種々の目的を持つ種々のレベルのユーザが想定できることがわかる。数式処理システムの開発に関してこれらの全てを満足する統一的システムを志向する方向と、個々の目的に合った特殊システムの構築の2 つの方向が考えられる。

従来の様に、特定のハイレベルの研究者の特定のニーズを前提としたシステムの設計だけでは、広範にユーザが拡大しようとしている現状に対応できない。

Ⅱ.　I／Oの改善

マイコン技術の劇的な進歩に支えられ、パソコンの高級化，ワークステーションの広範

な普及が始まろうとしている。ハードウェア的な入出力に関しては、キーボードより入力しキャラクタディスプレイにのみ依存する従来の入出力形態では不十分である。

特に柔軟な作業環境を確保するためには、マウスとビットマップディスプレイの導入は必要である。$\frac{d}{dx}$,$\int dx$等の様な数学的な表記法で処理結果を出力するために、ビットマップディスプレイとそのハードコピーは大きな役割を果たすであろう。2 表の3)の目的で数式処理システムを利用する際に、数式処理システム上で引用できる大規模な数学知識のデータベースが必要とされハードディスクは欠かせない。

それらの上にソフトウェアにより作られる作業環境として、

①アイコン= マウスの指向

②机上(またはノート上) の再現

③データの可視化

④多重ウィンドウ

等のAI用ワークステーション，ビジネス用ワークステーションの持つ機能を、数式処理システムも持つ必要がある。[1)]

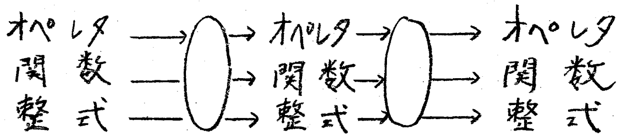
しかし、数式処理システムは上記の他のシステムに比較して入力に関してキーボードに依存する部分を多く持つと考えられ、入力に対する改善には解決すべき問題が多い。入力の再利用，部分的再利用は、この環境上では改善されようが、初期入力のキーボードよりの脱却には卓越したアイディアが要求される。[3)]

Ⅲ. 数理的なヒューマンフレンドリネス

従来の数式処理システムは整式に関わる処理を主たる目的として設計されているので関数，オペレタに関する取扱いに関して扱にくい面を持つ。筆者達がmuMATH上に作成した数式処理機能ICASは[2),4)]、1)に述べた数式処理システムに対する要求をできる限り取り入れたものである。

そこでは、オペレタ代数，関数の評価の抑止，独立変数の評価の抑止,SDIF,SINTの形式微分積分機能，柔軟な定義機能MAKERULE等が実現され、muMATHの持つ通常の数式処理機能と上記の解析的機能を融合して使用することができる。

従来の殆どの数式処理システムでは、関数，オペレタ，整式を入力すると出力は整式のみであるが、ICAS機能を持つmuMATHにおいては、出力として整数，関数，オペレタが出力できるので、



1図　ICASに於ける数学的量の入出力図

1 図に示した3 つの数学的対象を入力側のみならず出力側にも利用した柔軟なアルゴリズムが作成できる。そのためには、関数，オペレタを単なるLISP関数としてではなく、その属性ごとに異なった形で登録することになるが、このことが逆に評価の抑止を可能とし図の処理を可能にしている。

1 .Taylor 展開と知識の説明機能

$U(x) = SIN(X)$のTaylor展開を行うとき、ノート上では

$$U(X) = U(0) + U_X(0)X + \frac{1}{2}U_{XX}(0)X^2 + ---- \quad ①$$

$$U(x) = X - \frac{x^3}{3!} + \frac{x^5}{5!} + ---- \quad ②$$

の両方を使い分けるが、ICASにおいては

```
 DEFINE U(x),SIN(x),ENDDEF;
TAYLOR(U(x),x,0,5);
       ----> x + x 3/6 + x 5/120                ③
DISABLE(U(x));
TAYLOR(U(x),x,0,5)---->U(0)+Ux(0)*x+Uxx(0)*x  2/2+---   ④
```

と同じ $U(x)$ を用いて紙上の①，②を各々③，④と再現することができる。

Taylor展開に関する知識が④の形で取り出せたことになり、これは公式の表示に他ならない。（$U(x)$が定義された後でもこれが行えることに注意したい。）

Taylorは勿論、アルゴリズム（ プログラム） として関数定義されているが、人間の（ 用いる優れた） 記号法は①，④の形を見ただけでアルゴリズムを推察することができる。

```
TAYLOR(U(x)*V(x),x,0,5);                         ⑤
```

の様な、より複雑な場合でも事情は同じである。

通常のAIシステムにおいてはシステムの持つ知識を表示する特別のエディタが必要となる

が、ICASはライプニッツに始まる人間の記号法のメリットを継承しているので、通常の処理がそのまま知識の表現となる。しかも、④，⑤の様にその時々でのユーザの必要な形で場合に応じた知識が取り出せるので、柔軟な知識表現ともなっている。この機能を利用して、ユーザに数学知識やオペレーションに関して助言を行うことができ、数式処理技術を用いたCAI に対して新しい側面を拓くことが出来るが、これについては文献5 ,6　に詳述した。

2.非線型微分方程式の記号的[7]巾級数解法と『解析的アルゴリスム』

非線型常微分方程式の記号的巾級数解法について

$$U_{xx}^{3}(x) = U_{x}(x) + U^{2}(x) \qquad ⑥$$

を例にとり、ICASの特徴を説明したい。

$$U(x) = \sum_{i} a_i x^{i} \qquad ⑦$$

なる記号巾級数を仮定して、⑦を⑥に代入して　の各巾に対して係数の比較を行うと、

$$-a_1 - a_0^2 + 8a_2^3 = 0$$
$$-2a_2 - 2a_0a_1 + 72a_2^2a_3 = 0$$
$$-3a_3 - 2a_0a_2 + 216a_2a_3^2 + 144a_2^2a_4 - a_1^2 = 0$$
$$---\qquad ⑧$$

の a_i に関する記号代数方程式を得る。

これを a_0, a_1 が初期条件により、記号的，または数値的に与えられるとすると、⑧は逐次的に $a_2, a_3, a_4, ---$ に関する代数方程式を解くことにより求まる。このタイプの非線型微分方程式では a_3 以降については、一次代数方程式を解くことに帰す。この方式を『記号的係数比較法』と呼ぶ。

一方、全く異なった解法として次のものがある。⑤の微分方程式を形式的に微分して

```
3*Uxx 2*Uxxx          = Uxx + 2*U*Ux              ⑨
6*Uxx*Uxxx  2 + 3*Ux  2*Uxxxx = Uxxx  + 2*Ux 2 + 2*U*Uxx
------
------
```

を求める。これに $U_0=U(x=0),\ U_{x0}=U_x(x=0)$を代入して逐次評価すると、 $U^{(n)}_{(x)}\big|_{x=0}=$ $\underbrace{U_{x\cdots x}}_{n}(x)\big|_{x=0}$ が記号的に求まる。但し、⑨に関しては、 U_{xxx} についてsolve を用いて解く。次の項からはsolve を用いて単に一次代数方程式を解くこととなるので必ず解ける。

$$a_n = \frac{U^{(n)}(0)}{n!} \qquad ⑩$$

であるから、記号的に求まった $U^{(n)}_{(0)}$を用いて a_n を作り、⑦ に代入すれば、 $U(x)$ を記号的に求めることができる。この記号解法を『記号的代入法』と名付けた。

さて、⑨の第一式は⑧の第一式に、⑨の第二式は⑧の第二式に等しいことは、⑩を用いて ，⑨を書きなおすことで確かめ得る。

$$3U_{xx}^2U_{xxx} = U_{xx}+2UU_x \Longrightarrow 3\left(\frac{a_2}{2}\right)^2\frac{a_3}{6} = \frac{a_2}{2}+2a_0a_1$$

従って、等価な⑧，または⑨を係数比較法では、式の展開とCOEFF による記号的係数比較により求め、『記号的代入法』においては形式微分と記号代入により求めている。両者は勿論、全く一致した解を与えるが、前者は『代数的アルゴリスム』，後者は『解析的アルゴリスム』であることに注意したい。

さらに、

$$\overset{m}{\underbrace{U_{xx\cdots x}}_{n}}(x) = F(\underbrace{U_{x\cdots x}}_{n-1}(x),\cdots,U(x),x) \qquad ⑪$$

のタイプの非線型常微分方程式に対して、この記号的代入法が適用できることは明らかであろう。

$$U_{xx} = U_x + U^2 \qquad ⑫$$

の様なより簡単な非線形常微分方程式に関してはSOLVE を用いることなく単なる代入法により巾級数解を求めうることはすでに報告した。[8)]

一般に⑧を作る数式処理過程より、⑨を作る数式処理過程の方が高速であり、『記号的代入法』は記号的巾級数解法の処理の高速化にも繋がるものである。

さて、上記の『記号的代入法』が数式処理システム上で可能であるためには、

1)形式的記号微分(SDIF 法) が可能である。

2) $U_x,\ U_{xx}\cdots$等に代入が可能である。

の条件をみたす数式処理機能が実現されていればよいがICASはこれを持つので、代数的でない解析的なアルゴリスムの実現が可能となる。

紙の上でこの発想を思いついても従来の数式処理システムを用いたのでは、それを係数比較法に転換して数式処理のプログラムを作成する必要があった。ユーザにこの様なアルゴリスムの転換を要求する数式処理システムは、ユーザである数理研究者にとりヒューマ

ンフレンドリーではない。『代数的アルゴリスム』も『解析的アルゴリスム』も、その時々の発想や要求に応じてとれる数式処理システムこそヒューマンフレンドリーなものであり、ICAS機能はmuMATHを数理研究用のヒューマンフレンドリーな数式処理システムに転化することに成功している。数式処理システムはユーザの数学的な知識による問題の変換を必要としない様に設計されることによりヒューマンフレンドリに成りうる。

まとめ

ICAS機能により『代数的アルゴリスム』ではなく『解析的アルゴリスム』が数式処理システム上で実行できる様になったことの意義は大きく、従来、紙の上でしかできなかった多くの数理的戦略を自然な形でそのままのアルゴリスムで実行できる様になった。数理研究上のヒューマンフレンドリネスの1 つの典型と言えよう。通常の数式処理はComputer Algebra と呼ばれるが、ここで提言している方式　は　何と呼ぶべきであろうか。

〔 参 考 文 献 〕

1) 酒井，　簡易言語とエンドユーザーズ言語，昭晃堂　(1984)
2) 対馬，　小型数式処理システムとその応用，情報処理，27，379，(1986)
3) 佐々木，元吉，渡辺，　数式処理システム，昭晃堂　(1986)
4) K.Tsushima ,T.Sato ,OPERATOR Algebra Package ,SIGSAM BULL .(to appear)
5) TSUSHIMA et al. ,Teaching Mathematics by Computer Algebra ,MCSE'86,(1986)
6) 対馬他，数式処理を用いた微分の教育,CAI学会誌(印刷中)
7) 対馬，佐藤，　非線型常微分方程式の記号的巾級数解法，大阪電通大科学論集.(1986)
8) 対馬，佐藤，微分方程式のPADE近似解,REDUCE プログラミング資料　第3 集,435 (1986)