

## 数式処理ワークステーションの開発

----- 数式処理におけるヒューマンインタフェース -----

大阪電気通信大学 対馬 勝英 (Katsuhide Tsushima)

大阪電気通信大学 広田 健児 (Kenji Hirota)

## 1. 序

数式処理はその潜在的な可能性にも拘らずその普及は遅いものであった。歴史的にみて特定な分野における特殊なツールとして利用されたため、その発展の方向はバランスのとれたものとならなかった。極く少数の特権的ユーザの注文を意識したシステムの開発が行われてきたため、数式処理システムはヒューマンインタフェースに対する全般的配慮が欠けている。数式処理システムの優劣は特定の数理的な問題を解決できるか否かで決定されるとの価値観は数式処理の研究者の暗黙の前提となっているようだ。

一方、コンピュータとの対話的環境に関して『インタフェースの科学』<sup>1)</sup>とでもいふべき境界分野がたちあがり、その成果としてマルチウィンドウシステム、ハイパーメディアなどが開発された。そこにはユーザの作業、操作に関する洞察が反映されており、問題解決の道具がコンピュータの上で単に対話的に利用できるだけでは十分なインタフェースとはいえないとの強力な主張がある。

筆者は従来より数式処理システムのヒューマンインタフェースの改善を試みてきたが、<sup>2), 3), 4)</sup>それとヒューマンインタフェースに関する新しい動向との関わりについて述べてみたい。

## 2. 数式処理におけるヒューマンインタフェース

数式処理におけるインタフェースの研究は人間がノートと鉛筆を用いて数理的な操作と思考を行うプロセスの分析にはじまる。良好なヒューマンインタフェースを準備するにはコンピュータハードウェアの与える対話機能を前提としI/Oを改良するだけでは充分でなくコンピュータに支援された問題解決過程の全体を視野に入れた問題の捉え方が必要である。

数式処理システムの利用は研究者の特化した問題解決ツールに限定されたものではない。1図に数式処理の利用方式の分類を示したが、ユーザの種類や利用目的により要求されるインタフェースの内容と質は異なったものとなる。

1図を大別すると特定の一つの数理的な問題のみを数式処理システムを利用して解決するという研究者による利用法(これをR方式とよぼう)があるが、この場合ユーザは特定の狭い問題解決分野のエキスパートであり、良好なインタフェースに関する一般的な問題

利用方式	処理の種類	インタフェースへの要求	コメント
研究者に依る利用 (R方式)	特定の長大, 困難な処理	それほど強くない	ユーザは特定の目的の達成にのみ関心がある
一般のユーザによる利用 (U方式)	種々の一連の処理	柔軟なものが要求される	問題解決, 学習, 知識の検索, レポート作成等種々の要求を持つ
数式処理の利用 (E方式)	機能の借用	目的によるが柔軟さが必要	数式処理システムの特定の機能のみが使われる

1 図 数式処理システムの利用形態の分類

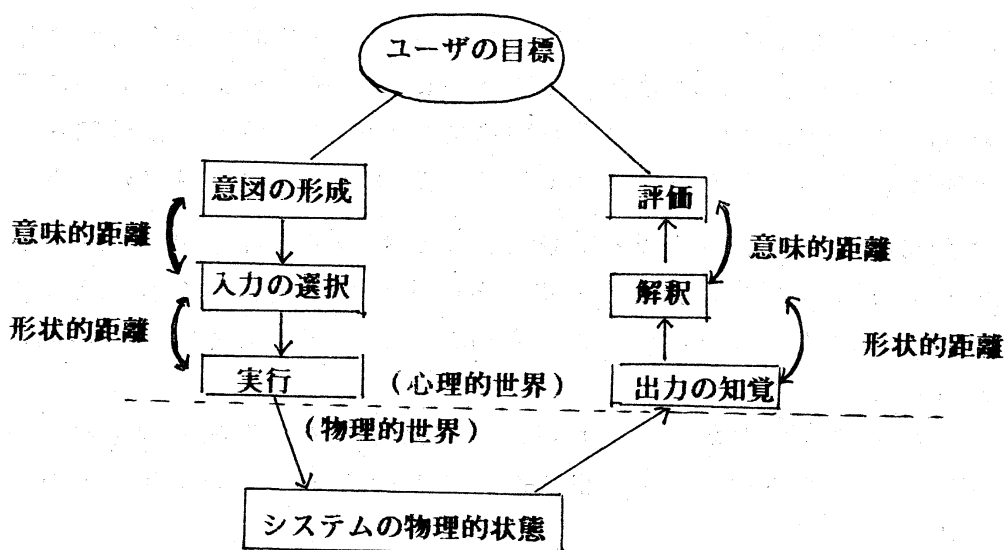
利用方式によりシステムに必要とされる機能とヒューマンインタフェースは異なり一つのシステムで全ての要求を満たすことは容易でない。

意識を持たない。

次にR方式ほど極限されていない利用法として数学公式集の替りに数式処理システムを使う(その整式としての形をしり, 数値的な値をしり, そのグラフを入手する)といった利用法がある。これは統合的な利用であり, この様な利用の可能なシステムは「統合化数学環境」と呼ぶべきものである。(これをU方式と名付ける)。この種の利用においては特定の処理機能の有無より使いやすさ, 判りやすさが要求される。この種のユーザは特定の数式処理機能が利用できるだけでは満足せず良好なヒューマンインタフェースの存在を当然のことと考えている。

また, 数式処理システムを利用して数学教育や数理的な分野での教育を行う分野がある。(E方式)。<sup>5)・6)</sup> この場合には数式処理機能の性能よりも, 利用のしやすさ, 自然な利用方式の可能であることが重要視される。(この種の利用は特定のシステムの中に数式処理機能を埋め込むという利用法も取られるがここではそれには触れない。)

これらの3つの分野に共通したインタフェースと各々に固有のそれがあり明確な分類学が必要とされる。



2図 DMIに関する図

人間とコンピュータの相互作用のモデルが示されているが、実行側と評価側での意味的距離と形状的距離を小さくすることと両側の相互参照がDMIに欠かせない。(この図は参考文献1の引用である。)

### 3. ヒューマンインタフェースのモデル

インタフェースに関する一般的考察<sup>1)</sup>は重要であるが、それは一般原理よりトップダウンにシステムの設計指針が獲られるといったものではない。現時点で最も進んだインタフェースに関するモデルはNormanによるDMI(Direct manipulation Interface)であろう。

これは3図に示したようにユーザの心理的世界と物理的な世界を区別してユーザの行動をモデル化したものである。

そこではユーザに適切なメンタル・モデルをあたえ得ることがよいインタフェースであるとの主張がなされている。

これに関してSchneidermanは

- 1) 関係のあるオブジェクトを恒に画面に表示する。
- 2) 複雑なコマンドによるのではなく、オブジェクトに直接働きかける動作やメニューによる選択により入力する。
- 3) 操作ははやく、可逆的で、結果は直ちにオブジェクトの変化としてみることができる。

を持つインタフェースがDMIであるといっている。<sup>1)</sup>

従来のコマンドによるインタフェースはユーザとユーザの関心を持つ対象の間にシステムが恒に介在している。これは対話メタファーによるインタフェースであり、ユーザは間接的にしか対象を操作しないし、世界を間接的に眺めているという感覚をもつ。一方、D M I では操作はモデル世界メタファによっているのでユーザは対象に直接、働きかけ、かつ世界を直接眺めているという感覚を持つ。その際にモデル世界メタファに必要な特性としてシステムからの出力の表現とユーザからの入力表現が重なりあえることの重要性が強調されている。システムから出力されたものがユーザの入力として再利用できることの重要性である。

これらがD M I の基本であり、それがユーザにとって心理的に快適なインタフェースを生む。M A C 上のハイパーメディアはこれらのD M I の一つのモデルを与えている。

#### 4. 数式処理そのものに関するD M I

我々がI C A S の概念を提起してから時間が経過したが我々が意図したものは3図に示す数学的操作における二つのメタファの距離を接近させるという試みである。これは数式処理におけるD M I の実現の過程であるが、数式処理に固有の事情もあり、詳細に分析してみたい。そもそも、近代科学の基礎を確立した解析学は記号法の成熟を前提としている。<sup>7)</sup>数式処理は特に記号法との関わりの強い分野である。ユーザの用いる記号法の変更を必要とする数式処理システムはD M I の発想とは遠いものである。また、数学的な作業は記号法とは別にユーザインタフェースの一つのファクタである。この様に数学で用いられている独特の記号法を前提としてD M I を設計する必要のある点に数式処理のヒューマンインタフェースに固有の難しさがある。数式処理機能を利用するために処理に関するメタモデルの変更を強制されるようではそのシステムはヒューマンフレンドリーであるとはいえない。

通常、数学的操作で扱う実体は5図の様に4つの階層に分けて捉えられる。整式の層が必要である様にも思えるが、数式処理に関してはこの層を除外することはできない。

数式処理はこの4つの実体に対して操作を施し別の実体をかえすことに他ならない。人間が紙上で行う操作においてはある特定の記号を何とみなすかは人間にまかされている。

$$A : B + C$$

1)

はオペラ、関数、整式、数値のいずれであっても使用でき人間は文脈によりそのいずれであるかを知って紙上の操作を進めていく。紙の上では実体は自動的に生成されないが数式処理システムは実体の自動生成の機能の故に利用される。

しかし、殆どの数式処理システムでは二つのオペラタ $O_1$ ,  $O_2$ の

$$(O_1 + O_2)^2 \longrightarrow O_1^2 + 2O_1O_2 + O_2^2 \quad (2)$$

なるオペラタ代数の結果は次の処理においてオペラタとしてオペランドに作用させることはできない。これは2で述べたDMIの要件を満たさない。

また、関数 $F(x)$ を $x$ で微分した結果は

$$DIF(F(x), x) \quad (3)$$

により得られるが、関数が定義されているとこの結果は整式となり、そうでないと再利用の効かない形の関数が返される。

また、関数 $F(x)$ と $G(x)$ が次のように定義されているとき、

$$F(x) = x^3 + 3 * x \quad (4)$$

$$G(x) = \text{Sin}(x) \quad (5)$$

を実行すると

$$Fxx(x) + Gxx(x) + 5 * F(x) + 5 * G(x) \quad (6)$$

の様な関数(微係数)としての出力は得られず

$$5 * x^3 + 4 * \text{Sin}(x) \quad (7)$$

の様な整式としての出力しか得られない。

次に関数 $H(x)$ が上述の $F(x)$ と $G(x)$ により

$$H(x) = F(x) + G(x) \quad (8)$$

と定義されているとき、

$$(H(x) + 3)^2 \longrightarrow (F(x) + G(x) + 3)^2 \quad (9)$$

$$\longrightarrow F^2(x) + G^2(x) + \dots \quad (10)$$

を出力することもできない。

この様に関数に関数に対応させる処理を複数回実行した後でその結果を整式として評価することは殆どの数式処理システムでは実行できない。通常の数式処理システムは整式としての出力を想定して設計されていることがこれらの例よりも理解できる。

この様に人間が紙の上で実行可能な処理の全てが数式処理システムの上で実行できるわけではないことは数式処理システムの初心者でも大いに感ずるものであり、操作できる実体について制限を課すシステムはけっしてヒューマンフレンドリでない。(R方式のユーザよりU方式のユーザの方が全ての操作が数式処理システムの上で実行可能なことを望んでいる。)

また、2)に示した様に処理の出力が次の入力として利用できないことはDMIとしての基本的要件を満たしていない。

我々はICASにおいて上述のいくつかの課題を解決したが詳細については4と参考文献3に依らたい。

ここで述べた様な欠陥は個別にテクニックを弄すると局地的に解決可能な場合が多いが、そのことは我々が I C A S において開発した様なインタフェースの改善が不要であることを意味しない。人間が (1) を行う際にメンタルモデルの変更を迫られることはなかった。I C A S もまたその種の変更を最小限に押さえたいとの意図で開発されている。我々の開発した I C A S においては微分演算を前置型微分演算子を用いて記述できる。

$$\left( \frac{d^3}{dx^3} + 3 \frac{d^2}{dx^2} + 2 \frac{d}{dx} + 5 \right) U(x) \quad (11)$$

は

$$(D x x x + 3 * D x x + 2 * D x + 5) . U (x) \quad (12)$$

と入力することで実行可能である。一方、微分を関数としてしか記述できないシステムでは

$$D I F (U (x), x, 3) + 3 * D I F (U (x), x, 2) + 2 * D I F (U (x), x) + 5 * U (x) \quad (13)$$

と入力する必要がある。(ここで D x は微分演算子である。)

さらに

$$\left( \frac{d^2}{dx^2} + \frac{d}{dx} \right)^2 U(x)$$

は I C A S では

$$(D x x + D x)^2 . U (x) \quad (14)$$

と記述できるが、通常のシステムでは

$$D I F (U (x), x, 4) + 2 * D I F (U (x), x, 3) + D I F (U (x), x, 2) \quad (15)$$

とユーザのシステム外での処理を要求することになる。

記号法として (14) の方がすぐれているのみならず (15) よりも (14) の方が U (x) に対して直接の操作をしているとのメンタルなイメージをもてる。この意味で前置型記法のような記号法それ自身が D M I であることがわかる。

$$g(x,y) = x^2 + y^2 + \left( \frac{d}{dx} (3x + \sin(x)) \right)$$

$$\hookrightarrow 3 + \cos(x)$$

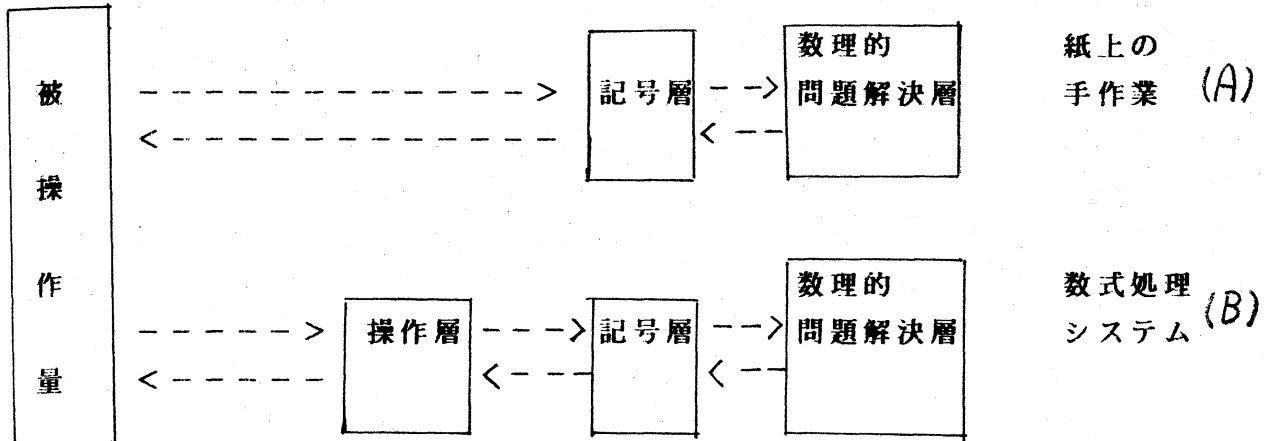

---


$$\text{Taylor}(\dots) = 1 + \dots$$

$$\rightarrow g(x,y) + 1 + \dots$$

4 図 ノートの一例

ノートの上では式の移動，式の命名，演算の手作業，記入の位置決め，全体の一望等が半ば無意識的に行われ，数式処理システムを利用する際にもこの事情はメンタルモデルの形成に影響を与えている。



3 図 数学的な作業と心理

数式処理システムを用いると出力の認知には敏感になるが紙上では殆ど意識されない。また，入力については紙上では記号操作そのものが意識されるが，その位置決め，移動，消去，同等性のマーク化等は無意識に行われ，操作層は心理的には無視できる。

## 5. 操作に関するDMI

数学を用いた問題解決を行う時の人間の作業をモデル化することにより、数式処理システムに要求されるヒューマンインタフェースの設計指針が獲られる。(この過程は自明のことであるとの立場にたつと数式処理システムのそれ自身のもつ機能の拡張しか視野に入ってくる。その為に特定の機能の強化のみが数式処理システムに関する研究であるとの狭い価値観に陥りやすい。)

3図に示すように人間は数式処理システムを用いた数理的問題解決においていくつかの質の異なったレベルでの思考と作業を展開する。紙と鉛筆をもちいるときはA図の様に二つの過程を意識することが多く、コンピュータと対話して問題解決をする場合はB図の様な3つの層を意識した作業をしていると思っている。A図において作業的な層が意識されないことは人間が紙の上での鉛筆を用いた筆記、記号の囲い込みの操作や紙面上の段落の変更、紙面上の記述の順序等の認知を殆ど無意識に行えることに起因している。

一方、数式処理システムを用いた場合に上述の紙面上の操作に対応した操作をキーボードとスクロールする画面といった貧弱な環境のもとで行うため図に示した種々の管理を強制されるので作業層を強く意識することを余儀なくされる。このような環境は3で述べたDMIとは隔たった作業環境であり、より多くのユーザの利用を期待するならば改善の必要がある。

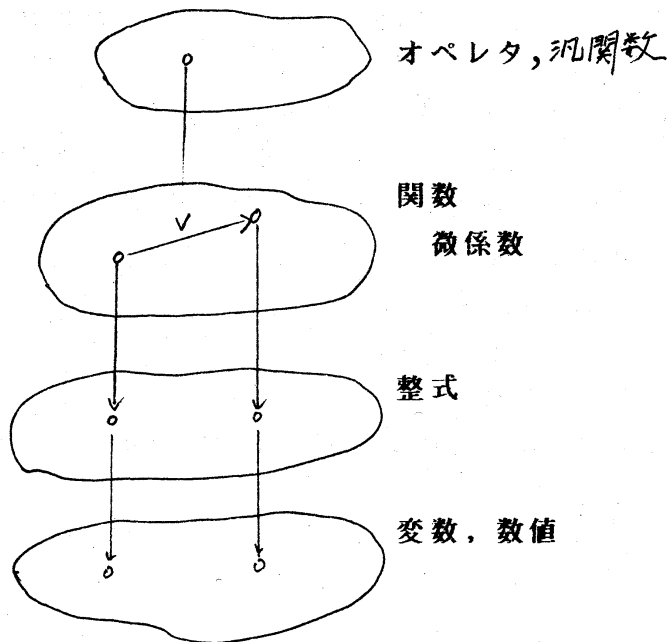
4図にノートの一部を示したがそこには数個の整式が記述されている。それらの相互の関連を視察し、部分の間の同等性の認知、整式の間論理的順序づけ、部分の注視、整式に対する操作の実施、全体の概観等多くのことがノートの上で実行される。我々はこの環境に慣れきっているため、これを分析的に捉えることは殆どなされない。観点をかえると紙上の操作においては永年の修練により一つのメンタルモデルを強制されているといえなくもない。

## 6. ICAS<sup>2),3),4)</sup>

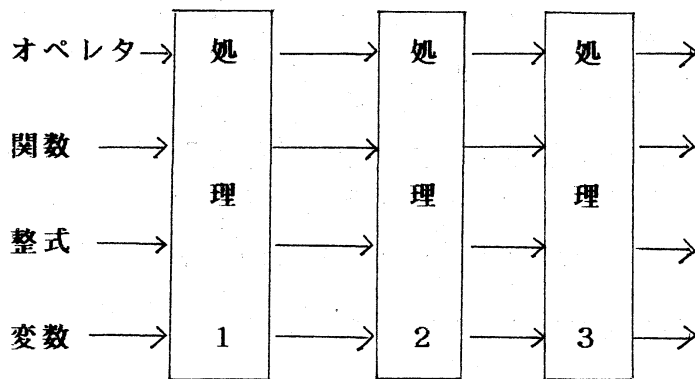
我々は微積分法、オペレタを多用する数式処理における操作性の改善と機能の拡大を意図して数式処理機能ICASを作成した。最初はmuMATHの上で開発したが現在ではLISP, PROLOG, SMALLTALK等の多くの言語の上で実現し、パソコン、ワークステーションの上で利用可能である。(機能の詳細は参考文献によりたい。)

- 1) オペレタの前置的適用を可能とした。
- 2) 出力されたオペレタ、微係数が見た目でなく実体として以後の処理の入力として利用できる。
- 3) オペレタ、関数、微係数、変数の4つの数学的実体の評価の抑止を任意のレベルで行うことができる。
- 4) 1), 2)に関連してMAKERULEと呼ばれる柔軟なルールの登録機能を持つ。





5 図 数式処理における4種の数学的実体



6 図 ICASにおける数学的実体の連続的操作

ICASにおいては4種の数学的実体を明示的に扱えるので、ある操作の出力は次の操作の入力として利用できる。これはDMIとして望ましい機能である。

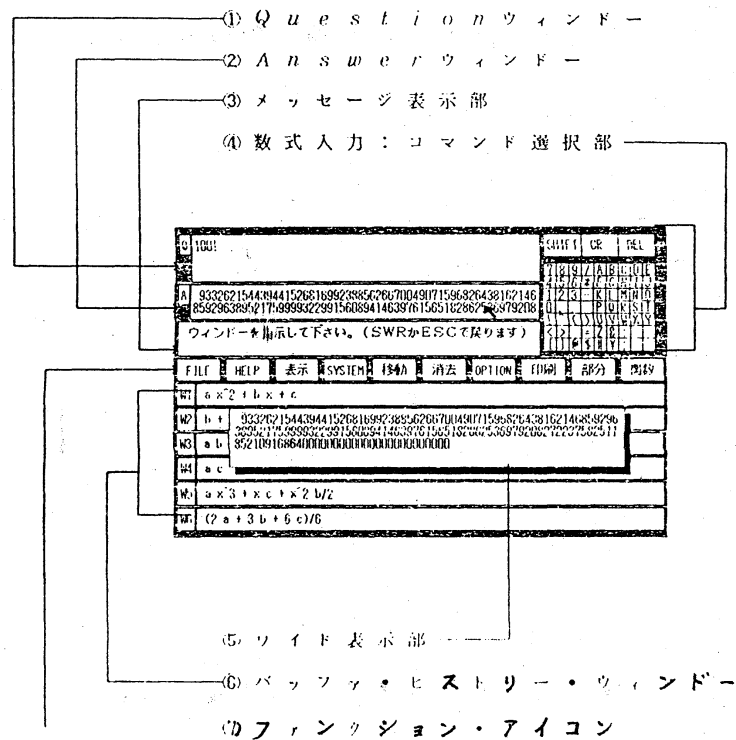
5) 2) に関連してオペレタ代数が可能である。

等が I C A S の特徴であるが、I C A S を利用したユーザの共通の感想として

- 1) 問題の解決までの全体としての時間が短縮された。
- 2) 微積分に関する知識データベースとして利用している。
- 3) オペレタ、関数について従来のシステムでは不可能なことが実行できる。
- 4) 従来のシステムとちがって出力を次の入力に使えるので処理の流れの一貫性が保て使用していて快適である。

等がある。これをみても I C A S が分野は限られていてもヒューマンインタフェースの改善を意識したシステムであることが判る。

I C A S は muMATH の対話方式を踏襲しており、I/O についての改良はおこなっていない。数式処理システムそのものの広い意味での記号法の改善がヒューマンインタフェースの改善になることを I C A S は示している。



7 図 INTCAS の標準画面

ユーザはつねにこの画面上のアイコンとそのサブアイコンと対話する。

## 7 数式処理ワークステーションINTCAS<sup>8),9)</sup>

ICASは4で述べた数学的な記号法と数学的な実体に関するDMIの実現であった。一方、5で述べた対話作業に関する改良もまたDMIの実現に欠かせない。

我々は数学教育における利用をその主たる目的としてICAS機能の利用できる数式処理ワークステーション機能INTCASをmuMATHの上に構築した。?図にしめす標準画面上のメニューとマウスを用いて対話することで数式処理をおこなうが以下の特徴をもっている。

- 1) 全ての操作はマウスを用いて可能である。
- 2) 画面上に6箇のスタックを持ち、数式のこれらの間で移動、抜取り、追加がピジブルにおこなえる。
- 3) 2)のスタック以外にディスク上に数式のためのバッファを持つ。
- 4) 長大な数式に関してはその為のウィンドウを用いて対応する。
- 5) すべてのICAS機能が利用できる。
- 6) ユーザは固定された標準画面と対話することで作業が進行する。
- 7) 数式の移動、消去、抜取り等をマウスを用いて行える。
- 8) オペレタの移動、抜取りが可能で、かつ、その結果を任意のオペランドに作用させることが可能である。

### INTCASのユーザの感想は

- 1) 小規模の利用に関しては非常に使いやすく、ICAS以上に全体的な問題解決の時間の短縮が可能である。
- 2) ICASのプリティプリント機能と併用すると紙と鉛筆を用いずに数学がやれる
- 3) コマンドの教育をしなくても数式処理機能が目にみえる形で利用できるので、教育現場での数式処理利用の可能性が見えてきた。
- 4) 恒に固定した画面と対話しながらマウスを用いて操作が進行するので変数管理、ヒストリーを意識した操作が不要になり直接、数学を意識させることができる。
- 5) 『関数』アイコンを用いるとコマンドの入力が不要になり使いやすい。

等がある。後述するようにSMALLTALKの様な環境においては多くのスクリーンを開いて作業をすることが一般的であるが、INTCASは一枚の主画面との対話を想定して設計している。勿論、技術的には多くのスクリーンを開く方式をとることはそれほど難しいことではない。研究者のツールとしては多くのスクリーンを開く方が便利ではあるが4)にのべたような長所を保存しつつ拡張を試みる必要がある。特に、教育においては無方針な機能のみの拡張は混乱をもたらす可能性もあり、実践による結論が必要である。

## 8 統合化数理環境

DMIに関して注目すべき数式処理システムとしてDerive<sup>10)</sup>が出現した。これは全ての操作をアイコン選択により行うことを基本としたインタフェースを採用している。

例えば

```

X3 + 2X をXで微分するには
      Calculus
      Differentiate
EXPRESSION X3 + 2 * X
VARIABLE   X
      Simplify

```

とする。これは操作をアイコンで選択し、次に被操作量を入力する方式である。左の文字はシステムの指示であり、右はユーザの入力である。

または

```

      Author
      X3 + 2 * X
      Differentiate
EXPRESSION # C/R
VARIABLE     X
      Simplify

```

としてもよい。これはオペランドを先に選び、次にアイコンにより操作を選択する方式である。勿論、通常のシステムのように

```

      Author
      DIF (X3+2*X,X)
      Simplify

```

の入力も許されている。

操作のアイコン化の方式が成功するには操作に対応したアイコンの階層化が必要となるが、ある程度、それは成功している。

単なる操作として見るならば上述のDeriveのインタフェースはDMIとして望ましいといえる。しかし、数学的な記号法である

15)

とは異なっており、ここまでの革新を期待しないユーザも多いだろう。1図に述べた用途

によってはこのインタフェースは有効である。

ある整式を展開して、微分し、それを代数的に解くといった一連の操作を行う場合、DERIVEは快適である。画面上に被操作量が表示され、それに対する操作をアイコン選択するだけで一連の操作が進行していく。そのとき、ユーザは記号法を忘れて操作に対するメンタルイメージのみで作業を進行させているのであろう。

Derive は二次元出力の形で見やすい形の対話結果を出力するので処理はメニュー、アイコンの選択で、結果は数学的な記号法に従ったものを用いるという設計がなされている。

この種のインタフェースの一般化は比較的容易であるが、記号法と操作の距離をどのように評価すべきかの結論がまず必要である。

次に良好なインタフェースを持つMACの環境のもとでのMathematica<sup>11)</sup>を例にひき数式処理システムのインタフェースについて述べてみたい。

Mathematicaの一般的な特徴として

- 1) 高度の機能を持つ数式処理が行える
- 2) 数式処理と数値計算を連続して行うことができる。
- 3) 強力なグラフィック機能を持つ。
- 4) APL, LISP, PROLOGに似た対話型利用ができる。
- 5) プロシジャがC, SMALLTALKと同等に記述できる。
- 6) 強力なパターンマッチ機能とその柔軟な制御機構を持つ。

等があるがここではその持つヒューマンインタフェースについて考えてみたい。

Mathematicaの入力はDeriveのような革新的なものではない。しかし、MACの上のシステムではプルダウンメニューによりファイル管理、エディティングが行えるので再入力、入力の一部変更、抜き取り、消去、コピー、ペースト等が自由に行える。これは紙の上で人間の行う特定の項の囲い込み、アンダーラインの添付、部分注視とよく対応するが、実体の移動が可能であることとそれらの再利用、再評価が可能である点がハイパーメディアの利用により生じた特徴である。MACそのものがDMIの典型でありその特徴が数式処理システムに継承されていると言える。しかし、Mathematica ICASのような記号法に関する新しい意味を持たず、INTEGRAL, Deriveのような操作に関する新しい提言を持っている訳でもない。しかし、Mathematicaを使用してみると著しく快適なインタフェースを利用しているとの実感を持つ。この実感は

- 1) スクロールバーにより全ての処理結果を見ることが保証されている。
- 2) 不要な入力と処理結果はプルダウンメニューにより処理単位で棄却できる。
- 3) 入力と処理結果はノートブックと名付けた単位で保存でき、かつ、再評価が可能である

これらの機能は数式処理システムとエディタとの結合の結果であり、新しいものではないとの意見をきくが部分的に存在した機能が統合化されて、ユーザにDMIを使用してい

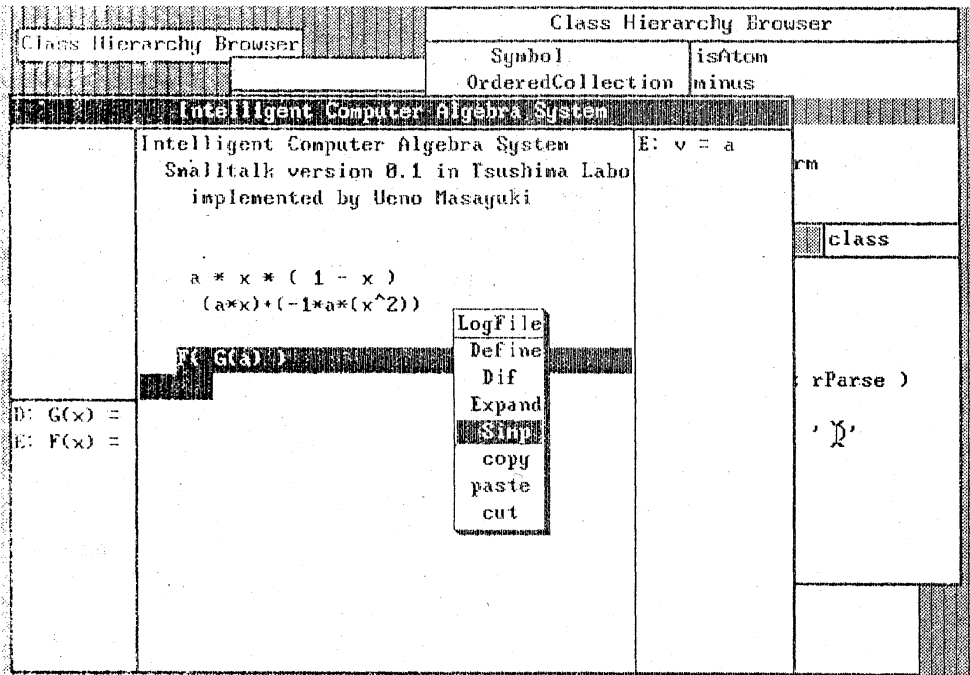
機能	muMATH	ICAS	INTCAS	Derive	Mathematica
実体の評価 の抑止	X	O	O	X	(O)
オペレタ代数	X	O	O	X	(O)
マウス	X	X	O	X	O
マルチスク リーン	X	X	X	X	O
部分抜取り 移動, 挿入	X	X	O	X	O
数学操作の メニュー化	X	X	X	O	X
画面	スクロール	スクロ ール	固定画面	固定画面	スクロール バーによる 全域参照

図 数式処理システムのインタフェースに関する特徴

本論文で論じたいくつかの数式処理システムのインタフェースに関する特徴を示した。Xは不可能，Oは可能，(O)は等価的に可能であることを示す。

るとの感覚をあたえることに意義がある。このようにMathematicaには我々がヒューマンインタフェースの特徴として望んだ多くの機能が実現されており、ハイパーメディア上の数式処理システムの改善の方向を探るには重要な示唆をふくんでいる。

4図で示した紙上の作業感覚に近い形の操作が可能である点は特に注目に値するがこの点の詳細な分析は我々もまだ済ましていない。

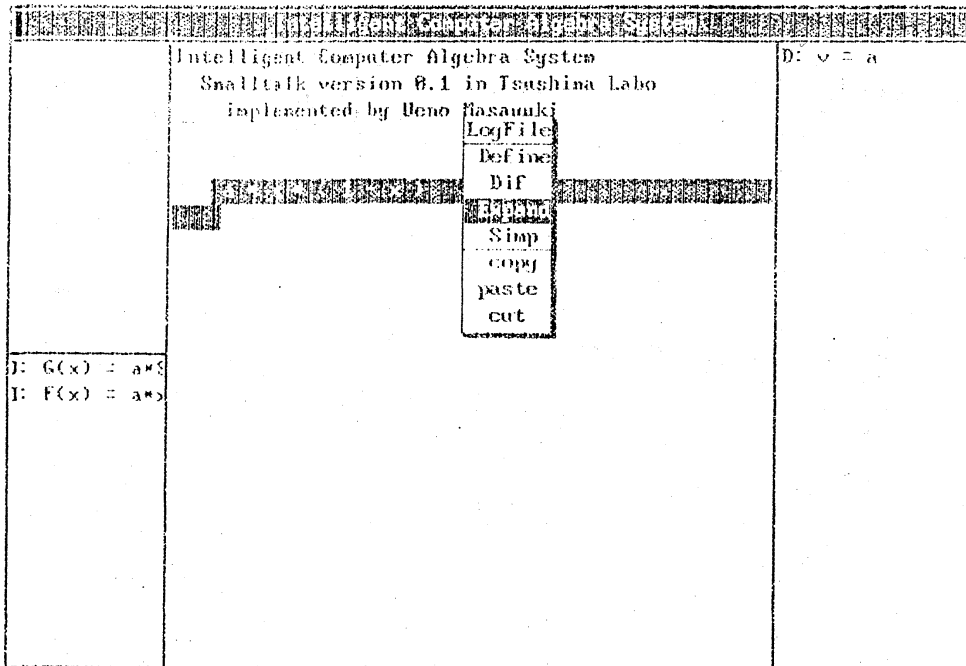


9図 開発中のSMALLTALK 上の数式処理システムの画面(1)

## 8. SMALLTALKをもちいた数式処理システムの試作

ICASの機能の利用を許しつつ, INTCASとは異なったタイプの数式処理システムの構築を試みた。マルチスクリーンとプルダウンメニューの使用はINTCASとは異なった作業環境を提供するが, 数式処理においてDMIを確立するために両者の比較検討は重要であると考えられる。現在, 7にのべた新しく登場してきた統合化数学環境のもつインタフェースの元での作業効率の分析を行っている。それをもとに1図の種々の用途に応じたインタフェースの作成を実践的に検討している。

8図に数種の数式処理システムのインタフェースに関する一覧を示したがICASの概念を利用できることを前提に操作のアイコン化を図る方向とハイパーメディア上で操作をアイコン化しないシステムの構築を当面の目標としている。(9, 10図に構築中のシステムの画面をしめした。)さらに将来的な目標としてDeriveのような記号法を離れた数式処理をハイパーメディア上で実現することは興味深い。これはE方式の利用において特に有効性を発揮する可能性がある。



10図 開発中のSMALLTALK上の数式処理システムの画面(2)

## 参考文献

1. 古川, 溝口, 「インタフェースの科学」, 共立出版, '87.
2. 対馬, 「小型数式処理システムとその応用」, 情報処理, 27, 379, '86.
3. 対馬, 「数式処理におけるヒューマンインタフェースの改善」, 第3回ヒューマンインタフェースシンポジウム論文集, 347, '87.
4. K.Tsushima, T.Satou, "OAP---Operator Algebra Package---", SIGSAM Bull.
5. K.Tsushima, H.Kaga, Y.Yoshimatsu and T.Satou, "Teaching Mathematics using a Computer Algebra", Proc. of MCSE'86, 407, '86.
6. 対馬, 加賀, 吉松, 佐藤, 「数式処理を用いた微分の教育」, C A I学会誌, 5, 59, '87.
7. 「ライブニッツ著作集」, 工作舎, '88.
8. 対馬, 広田, 「数式処理ワークステーションの開発」, 信学技報ET87-5, 19, '87.
9. K.Tsushima, H.Kaga and K.Hirota, "Mathematical Education Using a Workstation for Computer Algebra", Proc. of APCCE'88, '88.
10. SOFTWARE HOUSE, "Derive" Manual, '88.
11. S.Wolfram, "Mathematica", ADDISON WESLEY, '88.