

数式処理技術を活用した中学数学教育支援システム

慶応義塾大学理工学部 永田 守男 (Morio Nagata)
西村 元也 (Motoya Nishimura)
慶応義塾普通部 早川 肇 (Hajime Hayakawa)

要旨

科学や技術の理解にとって不可欠な文字式の内容は中学初年級の数学で教えられる。ところが、ここでつまづいて数学が嫌いになるひが多い。数式処理の技術を活用することによって、この部分の教育にコンピュータを利用できないだろうか考えたのが本研究の動機である。

従来からパソコン用として教育ソフトが市販され、また、人工知能研究の応用として知的CAIなども研究されてきている。しかし、それらは単なるゲームソフトの延長であるとか、これまでの教授法との馴染がわるい、または、コンピュータが学習を主導して生徒が受け身の姿勢で勉強してしまうといったような欠点があった。

そこで、ここでは、生徒からの積極的な働きかけで動くことを基本として、現行の教科書や教え方の中で役に立つコンピュータシステムを設計した。そして、その中に数学の先生の持つ経験と知識を取り入れ、数式処理や人工知能の研究成果を活用している。

本システムの特徴は、生徒の側から問題を設定したり解答の過程を入力できることであって、システム側がこれらを追いかけてその中にある誤りの検出とその原因を追求するアルゴリズムを備えていることである。しかも、各生徒に応じた記録を持つことによって、生徒それぞれに適した反応をシステム側から行なうように工夫した。

このようなシステムのほうから生徒の行動に合わせるためのソフトウェアを作成するには、難しい技術的な問題がたくさんある。ここでも、それらが十分に解決されているわけではない。本論文は、こうしたシステムを作るときの問題点を整理したものである。

1. はじめに

中学初年級の数学において文字式の問題が本格的に導入される。この概念がきちんと把握できるか否かは、その後の数学の学習だけでなく、コンピュータサイエンスをはじめとする広い範囲の科学や技術などの理解にも大きな影響を与える。物ごとを定式化するとか抽象的な考え方ができるようになるといったことの基礎に文字式の理解が不可欠だからである。これほどまでに重要な課程であるにもかかわらず、ここでつまづいて数学が嫌いになるひが多い。

したがって、この課程の教育にコンピュータを役立てることは大いに意味のあることなのだが、現在パーソナルコンピュータのソフトウェアの市場に広く出回っている教育ソフトウェアは、そのほとんどが単なるゲームかコンピュータ主導型のドリル帳でしかない。そして、教材の内容に杜撰なものも多く、この種のプログラムへの世間一般からの不信感を招いている。

一方、人工知能(AI)研究では、学習者についての心理学等の成果を利用した肌理の細かなシステムが作られてきた[1]。しかし、これらの多くがシステム主導型であって、学習者は受け身の姿勢で勉学することになる。あるいは、そのシステムのために特別に用意されたテキストを使いながらの特殊な学習法をとらざるをえないこともある。

こうした状況に鑑みて、学習者(生徒)の積極的な働きかけを主体にした、中学初年級の文字式の学習を対象とする教育システムを設計して試作中である。このシステムの概要と、これを作成するうえでの技術的な問題点を報告する。

本システムは次のような考え方を基にして設計した。

- (1) 現行の教科書や教え方の中で役に立つ「道具」を作る
- (2) 数式処理機能と人工知能の研究成果を活用する
- (3) 数学の先生の持つ経験と知識を取り入れる
- (4) 生徒ごとの個性に対応できるようにする
- (5) 計算機について何も知らない中学生が使えるものを作る
- (6) 生徒からの働きかけで動くことを基本とする

結局のところ、知的な能力を持つ「電子ノート」を数式処理技術を活用して作る事が本研究の具体的な形での目的である。

2. システムのあらまし

このシステムでは、特別なテキストや教え方を仮定しない。現行の教授法に従って単元の内容を教員が生徒に教科の内容を教えることを基本とする。その内容についての理解を深める補助手段として本システムを使う。したがって、クラスでの演習または自学自習時に利用することになる。

また、生徒からの積極的な働きかけを期待して作ってあるので、演習問題の提示等は生徒の側から行うことを基本としている。生徒側からの入力としては、

- (1) これから解くべき問題の種類
- (2) 問題
- (3) 解答またはその過程
- (4) システムへの諸要求

などがある。そして、システム側では、解答やその過程を調べることによって、誤りを検出して、各生徒のこれまでの学習状況に応じて

- (1) 誤りの箇所
- (2) 誤りの原因

などを出力し、生徒の要求に答えて

- (3) 解答

を示す。たとえば、一次式の計算の単元において、生徒が

$$2x+5x-6=6x-6$$

と計算してしまったら、ここで誤りを検出して警告を発するわけであるが、そのとき、その生徒が

- (i) 単なる計算間違い

をしたのか、それとも

- (ii) 同類項をまとめる概念がつかめていない

のかを過去のその生徒の学習記録から類推して、それぞれに応じた誤りについての注意または解答例を出力する。すなわち、(ii)がこの原因であると判定されている生徒が

$$5a-2b+3a-6b$$

という式を計算するときの解答を示してくれといえ、システムからは

$$\begin{aligned} &5a-2b+3a-6b \\ &=5a+3a-2b-6b \\ &=(5+3)a+(-2-6)b \\ &=8a+(-8)b \end{aligned}$$

$$=8a-8b$$

のように出力するが、(i)だと判定されている生徒が同じように要求すれば

$$5a-2b+3a-6b$$

$$=(5+3)a+(-2-6)b$$

$$=8a-8b$$

とだけ書く。もっと進んだ生徒に対しては

$$5a-2b+3a-6b$$

$$=8a-8b$$

だけでよいかもしれない。この例では、よく分かっている生徒に対して煩わしすぎる答えを出さないという意味しかないが、もう少し研究が進んだ段階で、生徒に応じた助言を与えることを計画している。

以上のような計算機とのやりとりを各單元ごとに繰り返せるようになっているのが、ここで報告するシステムである。これを作成するときの技術的な内容について述べるのが本論文のこれからの主たる目的になるが、その前に本研究の研究過程そのものについて説明しておく。研究の過程を箇条書きすれば次のようになる。

- (I) 中学初年級の文字式についての単元のまとめ
- (II) 各單元ごとの生徒が誤りやすい箇所と誤りの原因のまとめ
- (III) 上の結果を基にしたシステムの設計
- (IV) 実現
- (V) 実験
- (VI) 評価

このうちの (I)と(II)は、中学校の現役の先生方から知恵を拝借した。(III)から(VI)はサイクルを描いて研究が進むはずである。すなわち、(VI)の結果によって随時設計変更を行う。(V)は中学校の生徒に実際に使ってもらうことを予定しているが、現時点では、まだ実行していない。

3. 数式処理機能

中学一年の数学で学ぶ文字式についての内容をまとめ、これらに対応して本システムが用意している数式処理の機能を説明する。

3.1 中学一年の文字式の單元

現在の中学一年生のための数学のカリキュラムのうちで文字式に関する部分をまとめると次のようになる[2]。

(1) 文字式の表記規則

xの3倍($3 \times x$)は $3x$ と表記するといったような規則を学ぶ。

(2) 式の値

xが5という値を持てば $5x+3$ の値は28であるといった意味を理解する。

(3) 1次式の計算

項、係数、同類項といった概念を習得する。

(4) 等式

等式の意味と計算法を学ぶ。

(5) 1次方程式

1次方程式の意味と解法を習得する。

(6) 分数を含む式

分数を含む式についての計算が自由にできるようにする。

3.2 数式の入出力形と内部形

いま広く普及している CRTディスプレイ装置を使ってこのようなシステムを作ると、数式の入力と出力の形に関して不都合な点がたくさんある。たとえば、

$$(5*x+3)*(6*x^2-1)$$

のような式を入力させるようでは、

$$(5x+3)(6x -1)$$

と書くことを教えている単元の内容と矛盾する。乗算記号の省略は変数名を1文字とすることによって可能であるが、べき乗記号の省略は難しい。

したがって、最終的にはハードウェア等の改良を待つこととし、現実のシステムでは

(i) べき乗記号 (^)

(ii) 分数式の除算記号 (/)

のふたつだけは、中学生が実際に使っている記法と異なるものとなっている。

たとえば、

$$\frac{x+1}{3}$$

は

$$(x+1)/3$$

と入力する。

このようにして入力された数式の計算機内部での表現は、システムの記述にLispを使っていることもあって、演算子前置のLispのS表現となっている。たとえば、

$$(/ (+ x 1) 3)$$

のような形である。

3.3 数式処理

中学一年生の数学の範囲を対象としたときに必要となる数式処理は次のとおりである。ここでは、多項式が主なものとなって、有理式が若干含まれる。

- (1) 簡素化
- (2) 代入、置換
- (3) 交換、結合および分配の諸法則の適用
- (4) 単項式と多項式の四則演算
- (5) 分数式の計算
- (6) 等式の変形
- (7) 一次方程式の求解

これらの機能をLispで実現することは現在の数式処理の研究レベルからすれば容易なことである。むしろ、こうした機能を活用して生徒の誤りを発見し、その原因を追求することのほうが難しい。

学習者それぞれの特性に応じて誤りの発見とその原因の追求をおこなうためのアルゴリズムを中心にして、本システムの実現について次に述べる。

4. システムの実現

本システムを実現しているソフトウェアとハードウェアの環境とシステムの構成を紹介し、生徒の誤りを発見してその原因を追求する作業をコンピュータの中でどのように実現しているかについて説明する。

4.1 ソフトウェアとハードウェア環境

本システムの試作に利用しているのは、慶応義塾大学理工学部管理工学科に設置されているVAX11/750であって、実現にあたってはUnix(正確にはEUNICE)とFranz Lispの諸機能を使っている。また、標準的なCRT端末(VT100)を入出力として利用する。

このようなソフトウェアおよびハードウェアの環境は現在の中学校で手軽に使えるものではないが、近い将来に安価なLispマシンで強力な入出力装置の使えるものの出現が期待されるので、この構成は本研究のような目的に対して不自然なものではないと考えた。むしろ入出力機器としては、より高度で使いやすいものを仮定し、この点に関してここに示した環境を超えるものが使えない限り実用化が難しい。なぜなら、いまのままでは3.2に示したような不都合が生じるからである。

いずれにせよ、入力に関しては手書きの感じに近いもの、出力についてはビットマップディスプレイのようなものが必要になる。

4.2 システムの構成

現在試作中のシステムの概要を図1に示す。この中の長方形で囲んだ部分がソフトウェアとして作ったものである。

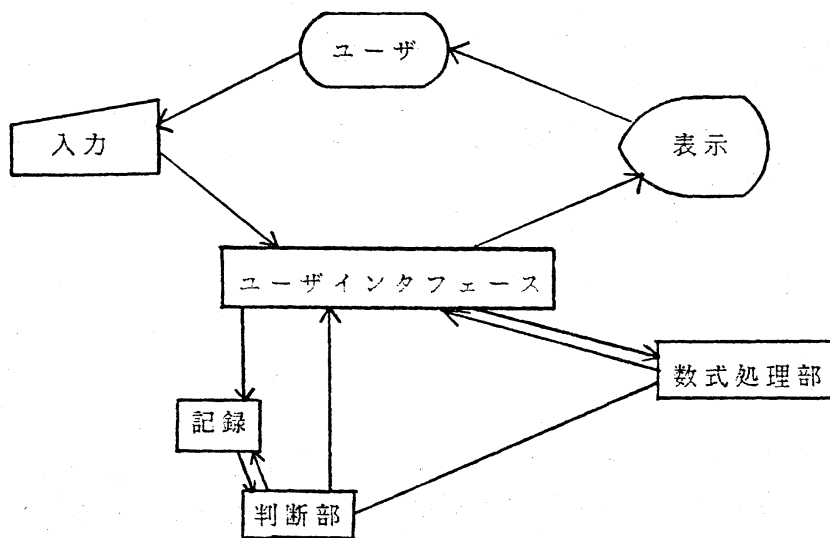


図1 システムの概要図

ユーザインタフェースは、入力された式を内部形に変更すること、内部形の数式を出力の形にすること、およびコマンドを解釈して実行することを受け持つ。コマンドの種類は少ないほどよい。現在用意しているコマンドは、

- (1) 問題の設定を知らせる。
- (2) 解答の提示を求める。
- (3) 誤りの箇所とその原因を知らせることを求める。

の3つである。これらをメニュー方式とすることも検討している。

数式処理部の持つ現在の機能は3.3で述べた。記録は各生徒の学習記録の要約である。判断部が本システムの中心となるもので、記録を参照しつつ生徒の誤りとその原因を追求する。記録のデータ構造と判断部のアルゴリズムを次に述べる。

4.3 記録のデータ構造

記録の部分のデータ構造は次のようになっている。このシステムを利用するたびに刻々とその内容が変わる。

```

((生徒1
  (单元11
    (回数111 誤り方111)
    (回数112 誤り方112)
    ⋮
    (回数11n1 誤り方11n1))
  (单元12
    (回数121 誤り方121)
    ⋮
    (回数12n12 誤り方12n12))
    ⋮
  (单元1m1 (回数1m1n1m1 誤り方1m1n1m1)))
(生徒i
  (单元21 . . . )
  ⋮
  (单元2m2 . . . ))
⋮
(生徒k . . . )

```

生徒によって学習の進度が違うので、どこの单元まで勉強しているかが異なる。さらに、どのような誤りを何回したかの履歴も生徒ごとにそれぞれ違っている。誤り方というのは、单元ごとに生徒がどのような誤りを犯しがちであるかを前もって整理したもののうちで、実際にその生徒がそれまでにしてしまった誤りであって、その回数とをペアにして記録する。回数を前にしてあるのはLispプログラムの都合による。こうした情報は、あとで述べる生徒の誤りを検出するアルゴリズムで使われる。

試作システムでは、このような単純なリスト構造として記録を整理しているが、近年のLispにあるデータ抽象化の機能やオブジェクト指向プログラムを活用することも考えている。こうしたことについては、試作システムによる実験を通して改良をすすめる。

4.4 誤りの検出

生徒の誤りの検出とその原因を追求するアルゴリズムの概要を示すまえに、こうしたシステムでは、生徒の入力の過程をトレースして誤りを見つけ出すこと自体がいかにかに難しいかについて簡単な一次式の計算の例を挙げて説明しておく。たとえば、生徒が問題とそれを変形した

$$2x+5x-6=6x-6$$

を入力したと仮定する。本システムでは、入力が必要な形であっても、項を入れ換えて標準的な並びにしておく、それを演算子前置のS表現に変換して内部形とする。また、項の中での定数や変数の並び方にも一定のきまりがある。その結果、上に示した入力は

$$(- (+ (* 2 x) (* 5 x)) 6) \quad (1)$$

および

$$(- (* 6 x) 6) \quad (2)$$

という内部形をもつ。(1)式から(2)式へ生徒は式の変形をしたわけだが、コンピュータの側では、(1)式のあと

$$(- (* (+ 2 5) x) 6)$$

とする。この式と(2)式とを比べて一致していないからといって直ちに(2)式への変形が誤りであるとは断定できない。さらに上の式を

$$(- (* 7 x) 6)$$

としたあと、この式がこれ以上の変形が不可能であって、(2)式もこれ以上の変形ができなく、かつ上の式と(2)式が一致しないことから(2)式への変形が誤りであることが分かる。

以上のプロセスを次の図2に示す。

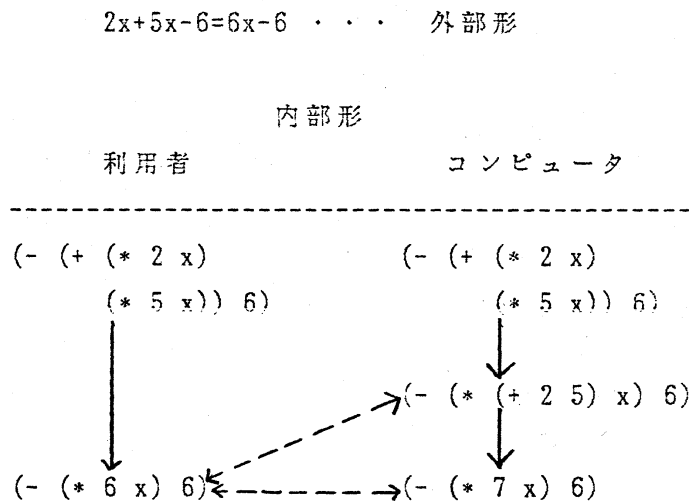


図2 誤りの検出過程

4.5 誤りの原因の追求

誤りのあることを見つけ出すことが難しいわけだが、その誤りの生じた原因（生徒の理解不足や計算間違い）をコンピュータ側から追求するのは、さらに困難である。ここでも簡単な例を挙げて、試作システムでこのために使っているアルゴリズムの概要を示す。

本システムには中学の先生方からの協力で集められた「知識」が系統的に整理されて格納してある。その内容は、たとえば

「 $10a+b$ を $10ab$ と書いてしまう」

とか、

「 a^n を na と混同している」

といったようなもので、生徒がどのような誤りをするかについての具体的な記述である。

この「知識」と「記録」を利用し、生徒からの入力を解析することによって誤りの原因を追求する。

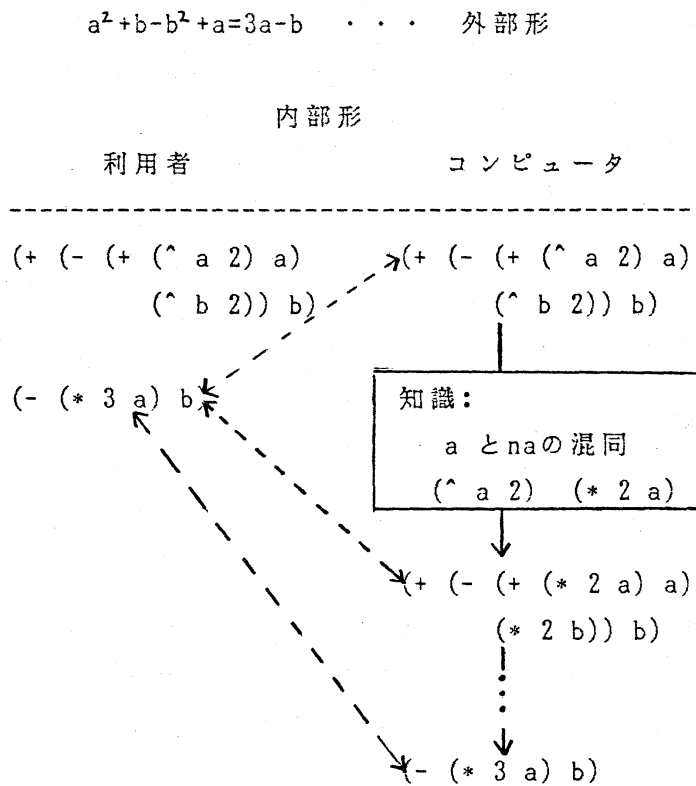
4.4 で述べた方法で誤りのあることを見つけると、その原因を次のような手順でコンピュータが推論していく。

ここで使っている最も基本的な手法は、「知識」の中から誤り方の候補を探り出して生徒と同じ誤りをやってみるといものである。たとえば、

$$a^2+b-b^2+a=3a-b$$

と生徒が書いてしまった場合を考えてみる。この式の左辺は、共通因数でまとめる以外に通常は変形できないはずなので、この式に誤りがあることは容易に発見できる。

このとき、「知識」の中から「 a と na の混同」をこの誤りの原因の候補として採用すれば、次のような内部形の変遷を経て生徒と同じ結果にコンピュータが到達する。



これによって、この生徒の誤りの原因がaの定義についての理解不足であったことが分かる。

こうした手順で誤りの原因を追求するときの最大の問題点は、誤りの原因を見つけ出すために要する手間が極めて多いことである。結局は、コンピュータが生徒と同じ誤りをやって同一の結果を出すまで、その原因を特定できない。さらに、「知識」の中に用意のない種類の誤りを生徒がやると、もはや原因の追求は不可能になる。

少しでも効率よく誤りの原因を見つけ出して適切な処置と正しい記録を作成するために、本システムでは生徒ごとの「記録」を活用する。すなわち、誤りの原因の候補として、最新の学習單元の中から充てはまりそうなものを選び、その中に原因となるものが見つからないときは、これまでに誤りの原因になっていた回数が多くかつ比較的最近に学習した單元の中から順に候補をあげて調べる。原因が究明できたら、「記録」の中で該当する誤り方の回数を増やす。

このようにして効率の向上を図ってはいるが、この手法自体が本質的に多くの検査を必要とするうえに、実際の生徒の誤りの中には誤り方が複合しているものが多い。したがって「組合せ爆発」を防ぐことができない。

従来のCAIソフトと異なった肌理の細かなシステムの作成を旨とする本研究では、このあたりの問題点も大きな研究課題である。

5. 数式処理と教育システム

数式処理アルゴリズムの進歩、数式処理システムの拡充、ハードウェアの低価格化と大容量・高速化などによって、数式処理の実用化が進展しつつある。それに伴い、各方面からの数式処理への期待と需要が大きくなってきている。多方面に渡る数式処理の応用が考えられるが、本研究は「教育」それも中学初年級の数学の一部についてのみ数式処理技術を使ったものである。このシステムの製作を通しての「数式処理と教育」に関する考察をここでまとめておく。

数式処理と教育との関係については、さまざまな側面から考えることができる。たとえば、一松教授の発言[3]のように、コンピュータで早くできるアルゴリズムは人間に教えても良いアルゴリズムであるという点に注目しても有益な論議が成立する。あるいは、現存する数式処理システムを活用する教育体系を考えることも可能である。しかし、ここでは、数式処理研究のこれまでの成果を利用して中学や高校レベルの教育にコンピュータを役立てる点に焦点を絞って考えることにする。

もうひとつ、ここでの立場を明らかにしておく。それは、「中学や高校でコンピュータを教える」ことでなく、「中学や高校の教育の補助としてコンピュータを使う」という立場である。別の言い方をすれば、コンピュータに合わせた教育ではなく、現在の教育を基本にして、これをより良くするためにコンピュータをいかに使うかを考えるということになる。

以上のような経緯の基で技術的な観点から考察を進めると、現状の計算機や数式処理システムには多くの解決すべき問題があることが分かる。高校レベルまでの数学を対象にしたとき、MACSYMAやREDUCEなどのシステムの持つ数式処理機能には問題がない。不定積分までができれば十分である。しかし、これまでの数式処理研究の成果だけでは、その点以外の問題が山積している。それらのうちのいくつかを次に示す。

まず、この論文でも既に指摘したように、数式の入出力の形である。これは、単に「教育」からだけでなく、真に広く数式処理システムが使われるようになるために早急に解決すべき課題といえる。

次に、計算機による数式処理の結果と人間の計算結果とが必ずしも同じ形にならないことも問題となる。不定積分についてのこうした例が[4]に載っている。形は違っていても数学的に同等であることが確かめられればよいが、それさえも困難なことが多い。

さらに、ここで示したようなシステムの作成に既存の数式処理システムの一部の機能を利用することは難しい。コンピュータサイエンスのさまざまな分野で数式処理を使うことの需要が増えていても、やはり同じ問題があって、現在のところはまだ十分に使われてはいない。数式処理システムがそれ自身で「閉じた」システムだからである。

6. おわりに

中学の数学教育を支援するためのコンピュータシステムを作成するときの考え方、試作システムの概要、現状での問題点等について述べた。いまのところ類似の考え方のシステムがないので、研究は暗中模索の状態を進めている。小さな要素としては参考になりそうな研究[5,6]もあるが、実際にこのシステムを作る場合にすぐに役立つような知見はない。とにかく試作システムを使った実験を早く行ない、フィードバックを繰り返しながらシステムの完成度を上げていく予定である。

参考文献

- [1] Barr, A. and Feigenbaum, E.A. (Eds.): The Handbook of Artificial Intelligence, Vol. 2, William Kaufmann, Inc., California, 1982
(邦訳, 田中・淵監訳: 人工知能ハンドブック, 第II巻, 共立出版, 東京, 1983)
- [2] 文部省: 中学校学習指導要領, 1977
- [3] 一松信ほか: パネル討論会「数式処理システムはいかにあるべきか?」,
bit, Vol. 15, No. 6, 1983, pp. 526-538
- [4] 金田康正: 話題提供—数式処理をめぐって—, 京大数解研講究録, No. 486,
pp. 195-211
- [5] Genesereth, M.R.: The role of plans in automatic consultation,
Proc. of 6th IJCAI, 1979, pp. 311-319
- [6] Brown, J.S. and Van Lehn, K.: Repair theory: A generative Theory of
bugs in procedural skills, Cognitive Science, Vol. 4, 1980, pp. 379-
426