

代数演算における指導システムの試作

長町三生, 畑本恵子, 伊藤宏司, 福場良之, 辻敏夫

広島大学工学部

近年の CAI 研究では、AI 技法の導入によって生徒の理解の状態をモデルとして表現し、生徒のもつ考え誤りを推論するような知的 CAI が開発されている。本稿では代数演算に必要な手続き的技能に関して誤りをもつ生徒のモデルを取り扱う。Brown と Burton の提案した「バグ」のモデルに基づいて、生徒が誤答する原因は、誤った演算手続きを追ったことにあると仮定する。本稿では LISP を用いた手続きネットワークによる生徒モデルを乗除算にまで拡張し、さらに、同定された生徒の誤りに基づいて適切な問題を選択して与えるようなシステムを作成した。

A development of the tutoring system for arithmetic skills

Mitsuo NAGAMACHI, Keiko HATAMOTO, Koji ITO, Yoshiyuki FUKUBA, Toshio TSUJI

Faculty of Engineering, Hiroshima University

Shitami, Saijo-cho, Higashi-Hiroshima-shi, HIROSHIMA 724, JAPAN

This paper deals with an automatically diagnostic model of an intelligent CAI for student's misconception of arithmetic skills. Underlying Brown and Burton's Buggy model, we assume the students don't make errors in arithmetic calculation, but they perform misconceptual procedure of calculation. This paper is a substantially expanded version of Buggy model to multiplication and division, implemented by LISP language based on a procedural network. The present model is able to select the appropriate arithmetic tasks for student's misconception according to the procedural diagnosis.

## 1. まえがき

教師から生徒に知識を伝達しようとするとき、集団教育では個々の生徒に対応した指導は困難であることが多い。一方、計算機技術の発達に伴い、コンピュータを応用した CAI (Computer Assisted Instruction) により、生徒個人の学習進度に合わせた教育を行おうとする試みが1960年代よりなされてきた。

初期の CAI は単に生徒に問題を提示し、それに対する解答の正誤を判定するだけのもので、これによる教育は画一的なものであった。しかしながら、生徒の理解度に合った個人指導を行うために教授内容の表現、自然言語処理、生徒モデルの構築に人工知能 (Artificial Intelligence: AI) の技法を取り入れた知的 CAI (Intelligent CAI: ICAI) システムが開発されている<sup>1)</sup>。中でも最近では生徒の考え誤りを推論するための生徒モデルの表現に力が注がれている。更に ICAI では、教授内容に依存しないような知識表現や教育戦略を確立することが重要な課題となってきた。

Brown and Burton<sup>2)</sup> は、縦書き加減算の誤り診断システム BUGGY の開発において、問題を解くために必要な手続き的技能を手続きネットワーク (procedural networks\*) を用いて表現し、生徒の思い違いを明確に決定することに成功した。手続きネットワークは手続き的技能を副技能 (subskill) に分解した副手続き (subprocedure) の階層的集まりである。

BUGGY では、誤りを持つ生徒のモデルは一部を誤った動作をする副手続きで置き換えたネットワークで表現される。しかし、実際に誤りを同定する際にはネットワーク上のすべての手続きを探索しなければならないので、2つ以上の副手続きを置き換えて表現する場合には、その組合せが膨大な数になる点に問題がある。

本研究では、教授内容に依存しないような枠組をもつ ICAI システムを目標とし、まず Brown らの手続きネットワークによる誤り診断の方法を乗除算にまで拡張する。更に同定した誤りの情報に注目し、誤りの履歴に基づいて次に提示する問題を制御することで、より効果的な教育を行うシステムを試作することを目的とする。

## 2. 生徒モデル

### 2.1 生徒モデルの役割

前章でも述べたように、生徒の理解の状態を計算機上に表現するのが生徒モデルである。初期のフレーム型 CAI において採用された数学的な学習確率モデル (stochastic learning model) は、ある刺激に対する特定の応答の確率をモデル化しただけであり、生徒が何を知っているか、あるいはどんな誤った知識を持っているかということは表現していない。

生徒の知識を表すために用いられる代表的な AI 技法は次の2つである。

#### 1) オーバーレイ・モデル (overlay model)

生徒の知識を教授内容に関する専門知識の部分集合としてモデル化する。専門知識の中で、解答履歴から生徒が習得したと推定される知識にフラグを立てるもので、SCHOLAR<sup>1)</sup>、WHY<sup>1)</sup>において採用されている。

これとは反対に、生徒が知らないと推定される知識にフラグを立てるものは差異モデル (differential model) と呼ばれる。

#### 2) バグ・モデル (bug model)

生徒の知識を、専門知識の混乱状態または脱線状態 (bug) としてモデル化する。習得した知識をもとにした推論法に誤りがある場合を表現することができ、SOPHIE<sup>1)</sup>、BUGGY<sup>2)</sup>において採用されている。ここで、生徒の誤りは次の3つに分類される<sup>3)</sup>。

---

\*この言葉は Sacerdoti (1977) が用いたものであるが、Brownらは彼の "plan" という考えよりもむしろ手続き間の制御構造を表す特徴において用いている。

- a) 知識の欠落
- b) 知識の誤り (誤った知識を持っている)
- c) 知識の適用の誤り (正しい知識は持っているがその使い方が誤っている)

オーバーレイ・モデルではa)の誤りしか表現できないが、バグ・モデルではすべての誤りを表現することができる。

## 2.2 手続きネットワークとその動作

代数の問題を解くためには、様々な手続き的スキル、つまり代数学の公式を適用すること、数字を読み取ること、あるいは右、左といった概念の認識などが必要とされる。問題を解く行為は、下位の手続きを副手続きとして用いる、手続きの階層的集まりであると考えられる。

生徒が問題を解いて誤った答えを得た場合、Brownらは、その原因は生徒が手続きをうまく追うことができないことではなく、誤った手続きを追ったことであると主張した。また、ある誤りを含む手続きをもつ生徒は、その手続きを副手続きとして用いる他の問題にも誤りを繰り返すことがわかった<sup>2)</sup>。

そこで彼らは、問題を正しく解くために必要な手続き的スキルの制御構造を明らかにするような表現方法として、「手続きネットワーク」を提案した。この方法では、人間の持つ副技能がネットワーク上の各ノードに対応する。

この方法では、ネットワーク中のすべての副手続きが正しく動作するとき、ネットワークは正しく問題を解く。規則的な誤りをもつような知識は、1つ以上の副手続きを誤りを含む副手続きで置き換えることによって表現される。この表現によるモデルは前節のバグ・モデルである。

彼らが示した加算と減算のネットワークを図1および図2に示す。

ここで加算問題 
$$\begin{array}{r} 43 \\ + 9 \\ \hline \end{array}$$

を解く場合を例にとって手続きネットワークの動作を説明する。

最上位ノード(ADDITION)は2数の加算を行う手続きで、各桁について左のノードから順に手続きを呼び出す。"METHOD1" (右の桁から左へと足してゆき、繰り上がりは次の桁の上の数のところに書くアルゴリズム)では、まず1桁目の和を求める(SUM A COLUMN)を呼び出し、これは"3"と"9"の和が"12"であるとい

う知識(ADDITION FACTS TABLE)を呼び出す。次に(ADDITION)は(WRITE DIGIT IN THE ANSWER)により"12"の中の"2"を答えの欄の1桁目に書く。次に(CARRY INTO THE TOP OPERAND)が繰り上がりの"1"を処理する。2桁目についても同様に計算を行い、処理すべき桁がなくなると終了し、答え"52"を得る。

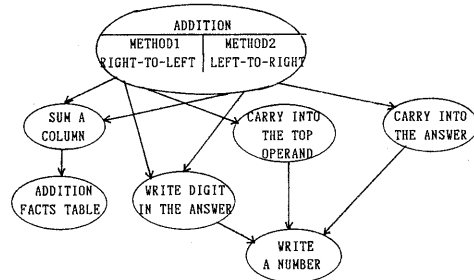


図1. 加算の手続きネットワーク<sup>2)</sup>

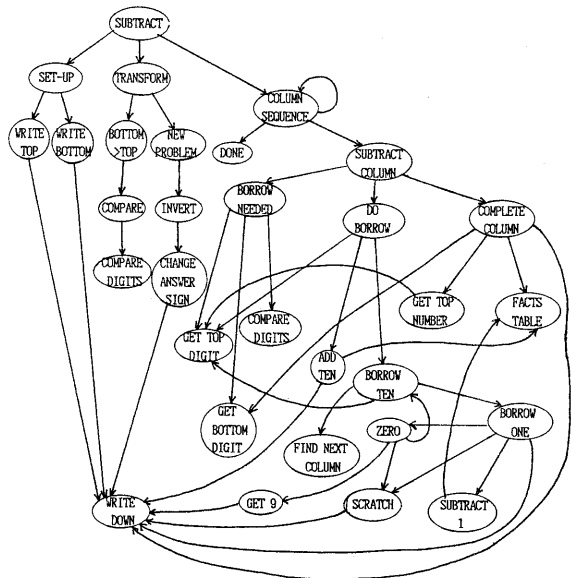


図2. 減算の手続きネットワーク<sup>2)</sup>

### 2.3 代数演算のための生徒モデルの構築

本研究では加減乗除の4演算におけるモデルを用いる。このうち加算は Brown らの手続きネットワークをそのまま、減算は彼らのものに多少修正を加えたもの(図3)を用いる。本研究において作成した乗除算のネットワークをそれぞれ図4および図5に示す。

図に示すように各演算は30個前後の副手続きから構成されると考えている。生徒の解答から誤りを同定する作業は、これらのネットワークの中でどの手続きがどの誤動作をするかを探索することである。したがって、生徒から解答を得るたびに全部の副手続きを探索するのは非効率的であるので、本研究では探索範囲を限定するような手法を考える。

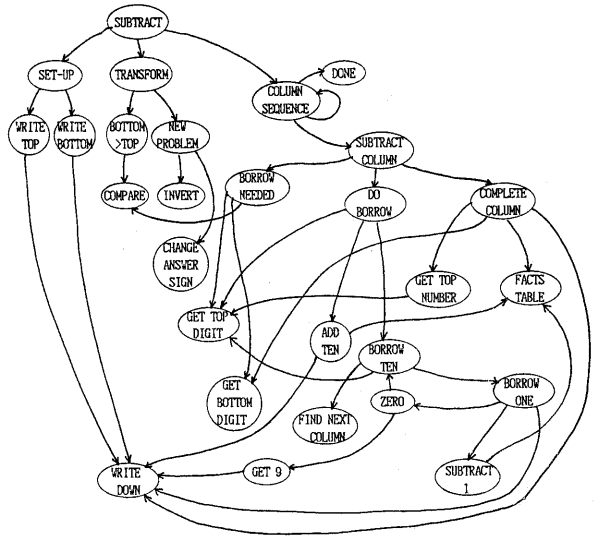


図3. 減算の手続きネットワーク

### 3. システム構成と問題の制御

#### 3.1 システム構成

図6に本研究で作成した ICAI システムの構成を示す。

本システムは、問題制御、問題選択、シミュレート、モデル生成、解答の照合、解答履歴記録の各機能を有する。またモデル生成に使用するデータ・ベースには、前章で説明したような正しい動作をする手続きネットワークと、置き換えるべき、誤りを含む手続きを蓄えている。問題データは小学校2~4年生程度の筆算形式の四則演算(整数)である。

以下、各機能について説明する。

#### 1) 問題制御

生徒の解答履歴により、問題のレベルを変化させる(レベルについては3.3節で述べる)。

ここでは、以下の基準で選択する問題の範囲を与える。

- ・ 現行レベルでの正答率が基準を越えると次のレベルに進む。
- ・ 同じ誤りと推定された誤答が現行レベルでの全誤答に対するある比率を越えると、その種類の誤りを生徒の弱点とみなし、その弱点についての基本レベル(3.3節で述べる)に移る。またこの制御によって移ったレベルで正答率が基準を越えると、元のレベルに復帰する。
- ・ 誤りが同定できない誤答が全誤答に対するある比率を越えると、前のレベルに戻る。

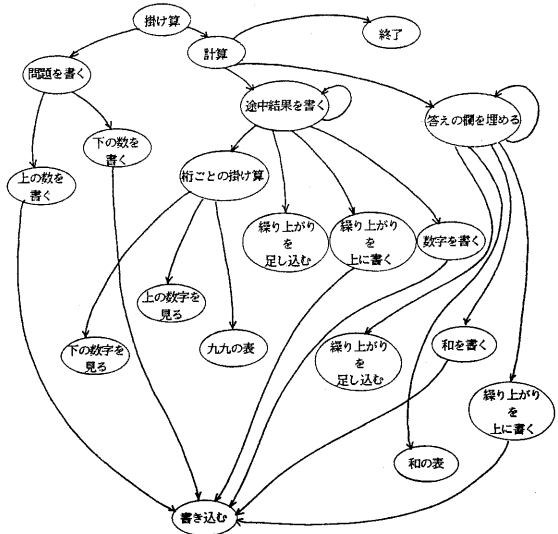


図4. 乗算の手続きネットワーク

## 2) 問題選択

1) の制御により与えられた範囲の問題データの中から問題を1つ取り出して CRT に表示し、生徒の解答入力を促す。

## 3) 解答の照合

問題に対する正答と生徒の入力を照合し、正誤によってメッセージを表示する。また、モデルにより得られた解と生徒の入力とを照合する。

## 4) モデル生成

生徒が誤答した場合、正しい動作をするネットワークの一部を誤りを含む手続きで置き換え、生徒の仮説モデルを生成する。

その中でシミュレートの結果と生徒の入力との照合が成功したものを生徒モデルとして採用する。

## 5) シミュレート

生成した仮説モデルによって問題を解く。

## 6) 解答履歴の記録

提示した問題と共に、生徒の入力した解答およびその正誤、推定された誤りを記録する。

なお、本システムの使用言語はLISP、使用計算機はPC-9801E（日本電気製）である。

### 3.2 誤り診断の方法

本システムでは、正しい動作をする手続きネットワークの中の1つの手続きを、誤った動作をする手続きで置き換えたものを生徒の仮説モデルとする。

前述の問題で

$$\begin{array}{r} + 9 \\ 42 \end{array}$$

とする生徒の誤りを診断する場合を例にとる。図1の加算の手続きネットワークの中で、(ADDITION (METHOD1))を繰り上がり処理の手続き(CARRY INTO THE TOP OPERAND)を呼び出さないような手続きで置き換えると「生徒は繰り上りを無視する誤りをもつ」という仮説が生成される。このときの手続きネットワークは図7のようになり、このモデルは図8のように動作する。これによって得られる結果"42"は生徒の解答と一致するので、この仮説を生徒モデルとして採用する。

### 3.3 問題制御の方法

本システムでは手続きネットワークに示される手続きの分解の仕方によって、データ・ベース中の問題を予め分類することによって、誤りの診断や問題の制御の効率化を図る。

本節では問題の具体的な分類方法と、それに基づく制御方法について述べる。

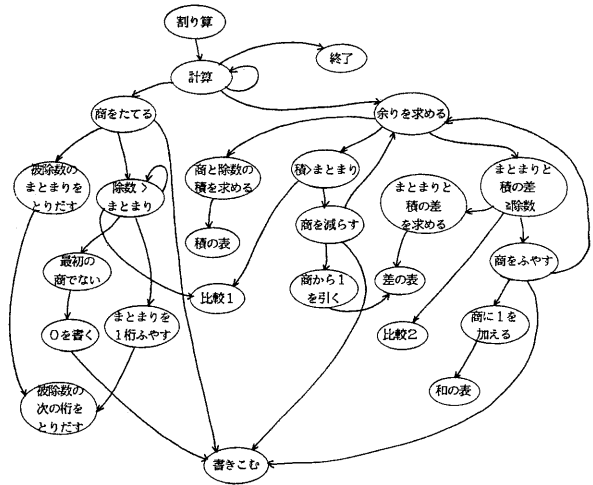


図5. 除算の手続きネットワーク

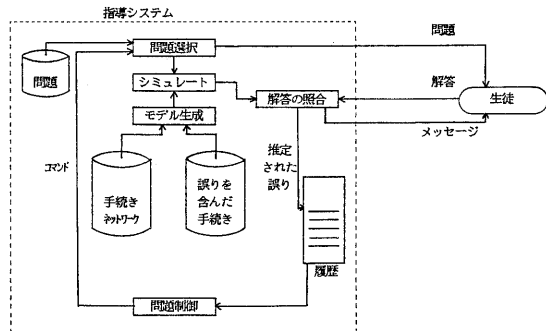


図6. システム構成

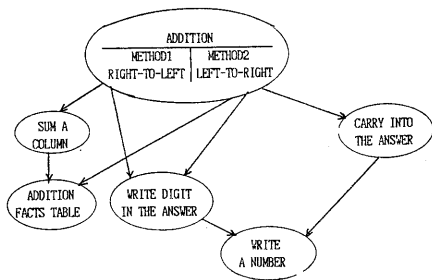


図7. 「繰り上がりを無視する」誤りをもつ  
加算の手続きネットワーク

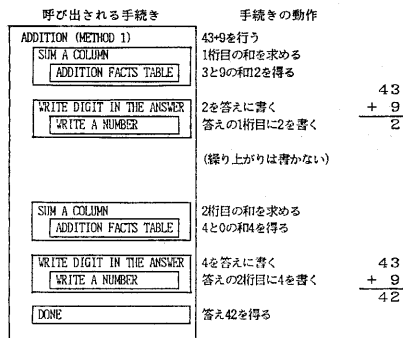


図8. 繰り上がりを無視するような  
加算ネットワークの動作

### 1) 問題の分類

加減乗除の筆算において、誤りをおこしやすい特徴には経験的に以下のようなものがあると思われる。

加算：繰り上がり

減算：上下の数の桁数の違い、繰り下がり、繰り下がりにおいて借りようとする桁の数字が0である場合

乗算：上下の数の桁数の違い、上の数の各桁に下の数を1桁ずつ掛けるときの繰り上がり、加算時における誤りを起こしやすい特徴

除算：商に0が立つ場合、割り切れない場合、減算時における誤りを起こしやすい特徴

これらの特徴と桁数から各演算の問題を分類した結果、

加算：25種類      減算：36種類

乗算：65種類      除算：49種類

となった。1つの種類の問題群を1つの選択範囲とする。問題制御の際の「レベル」は、同種の演算の中では問題の桁数が多いほどむずかしい、また桁数が同じ問題の中では前に挙げたような特徴を多くもつほどむずかしいという基準で予め設定している。さらに各特徴について桁ごとの基本レベルを設定している。すなわち、ある特徴についての基本レベルとは、その特徴1つだけをもち、他の特徴をもたないような問題群である。

### 2) 問題の制御

問題の制御は、1)で述べた分類の中からの種類の問題群を選択範囲とするかという決定である。問題制御に用いている判断を図9に示す。

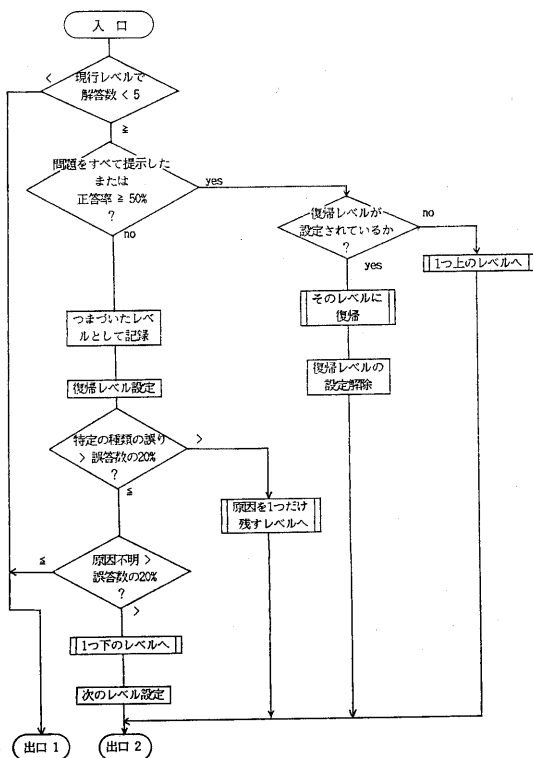


図9. 問題制御のフローチャート

この制御で問題の選択範囲を変化させるのは、次の3つの場合である。

- a) レベルを1つ上げる
- b) レベルを1つ下げる
- c) 基本レベルに移る

a)およびb)の場合には、予め順序づけたレベルをそれぞれ1つずつたどることによる。

c)の場合には、多く同定される誤りの原因に関する基本レベルに移行する。

### 3.4 実行の流れ

図10にフローチャートを示す。初期設定では生徒が初めてシステムを利用する場合と2回目以降の場合に分けて第1問のレベルを決定する。第2問以降は図9の判断によりレベルを決定する。

提示した問題に対する生徒の入力が誤答であるとき、仮説モデルの生成—シミュレートを繰り返して誤りの同定を試みる。このとき1つの仮説モデルを生成するために置き換える手続きは1つとし、生徒と同じ答えが得られるか、調べていない誤り手続きがなくなるまで続ける。

### 3.5 診断例

生徒の誤答に対する診断例を以下に示す。

Q.11は、 $756 \div 7 = 18$ としたものである。  
正解108を示したあと、この生徒の誤りは「商に0を書かなかった」と同定している。

# Q.11

7) 756

# kotae o irete kudasai 18

# machigai desu

# seikai wa 108

??? ayamari o shirabete imasu ...

# ayamari ga wakatta !!!

# anata no ayamari wa: (#shou\_ni\_"0"\_o\_kakanakatta)

### 4. あとがき

本研究では、Brownらの手続きネットワークによって生徒モデルを表現する方法を四則演算に拡張した指導システムを作成した。

モデル表現としての手続きネットワークは、生徒の誤りに対する表現力において優れているといえる。この表現によるモデルは、必要な知識の中で何を知っているか、何を知らないかだけでなく、誤った知識

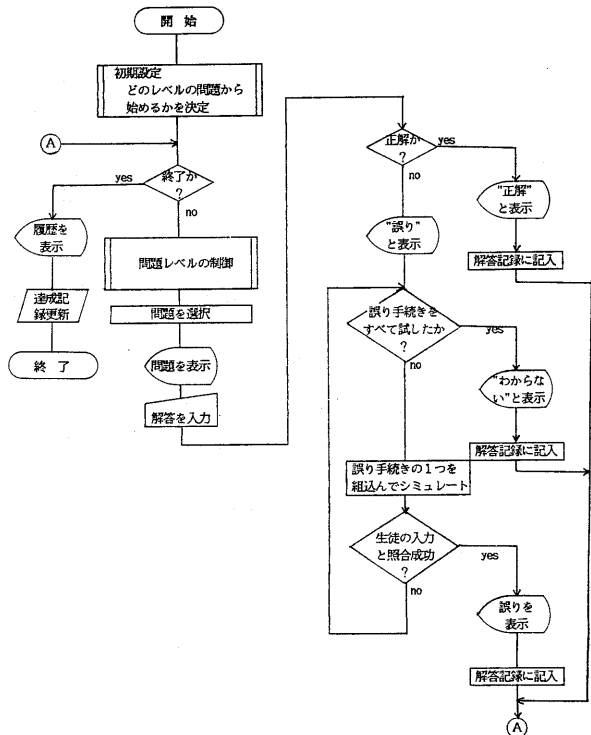


図10. フローチャート

を持っている状態や、正しい知識は持っているがその使い方が誤っている場合をも表現することができる。また、題材が明確な手続きで記述できるものである限り、その内容に関係なく、適用可能である。

また、本システムでは誤り探索の効率化および同定された誤りの履歴に基づく教育効果の向上を図るために、予め問題を術数によって分類し、各問題群を更に誤りを起こしやすい特徴によって分類している。Brown らの誤り診断ではネットワーク全体を探索するが、本システムでは提示した問題が含まない特徴に関する誤りを探索しないことでモデルの生成回数を軽減している。また、各問題群に対するレベルを設定し、その中で誤りを起こしやすい特徴1つだけをもつような問題群を基本レベルとすることにより、生徒の誤りの履歴から推定される弱点に焦点を当てた練習問題を与えるような制御を行うことができる。

問題点は、用意した誤りを用いて診断する方法では、それ以外の誤りには対処できないことである。また明らかな誤りが新たに発見された場合にも、それをプログラミングするのは容易ではない。これらの点を克服するためには、生徒の解答から誤りを学習する機能を持たせることが必要であろう。

## 5. 参考文献

- 1) Barr, A. and Feigenbaum, E.A. 編; 田中, 淵監訳: 人工知能ハンドブック第II巻, 共立出版, pp. 301-404, 1983.
- 2) Brown, J.S. and Burton, R.R.: Diagnostic models for procedural bugs in basic mathematical skills, Cognitive Science, pp. 155-192, 1978.
- 3) 河合他: 論理プログラミングと帰納推論による汎用知的C A Iシステム, 情報処理学会論文誌 vol.26, No.6, pp.1089-1096, 1985.
- 4) 文部省: 小学校指導書算数編, 大阪書籍, 1978.