

バグルールを抽象化する学習者の モデリングに関する研究

The study on student modeling for the purpose of abstracting bug-rules

— 因数分解の解法過程において —

In the process of solution on factorization

岡本 敏雄 小林 英忠
Toshio OKAMOTO Hidetada KOBAYASHI

東京学芸大学
TOKYO GAKUGEI UNIVERSITY

本研究は I T S における、学習者の理解構造をモデリングする方法について述べている。特に、合成されたバグルールの診断のありかたについて議論されている。具体的な題材としては、中学校数学における、因数分解を用いた 2 次方程式の解法を扱っている。システムの対話形式においては、学習者の回答に対して何らかの教授を与えることをせずに、学習者が如何なる解法を行ったかを認識するようにした。

キーワード： 知的 C A I I T S 学習者の理解のモデリング 診断 バグ合成の抽象化
L M S

1. はじめに

現在、知的 C A I または I T S に関する研究が盛んに行われている。それにより、その枠組みが明らかになってきた。しかし、実際に教育現場での利用を考えた場合には、まだ多くの問題が残っている。

我々はこれまでに N E W T O N₍₁₎, G E O M E X - II₍₂₎ などいくつかの I T S を研究開発してきた。それにより、対象世界を限定した範囲内で、I T S の構造モデルを蓄積している。

本研究は、対象世界として中学校数学における、因数分解を用いた 2 次方程式の解法の学習を支援するための I T S の構築および、適切な診断の構造を探究することがねらいである。

先行研究としては、Sleeman らの Leeds Modeling System (LMS)₍₄₎ があるが、LMS においては (診断) モデルと、治療的教育との分離をはかり、モデル生成のみを行う。またモデル生成においては、プロダクションルールを用いて、学習者の回答とのマッチングがとれるよ

うな推論が行われている。しかし前述したように、LMS においては、治療的教育を行うことを目的としていないので、学習者の誤りを同定するための診断機能を備えていない。

そこで本システムは、LMS による構造を参照し、学習者に対する診断のためのモデルを生成し、それを利用して、システムが適切な診断を実行できるメカニズムの構成が中心課題である。

2. 目的

本研究の目的は、I T S における学習者の理解構造をモデリングすることである。特に、合成されたバグルールの診断に中心課題を置いて、システムを構築する。

具体的には、中学校 3 年生での指導内容である因数分解、およびそれを用いた 2 次方程式の解法における基本的な式変形、演算技能を理解、習得させるための I T S の構築である。それは、様々な学習者の回答、解法過程を認識し、それをうけて、学習者の誤りを同定し、適切な診断

を与えて、学習を支援しうるシステムの構築である。

本システムは、2次方程式の解法においては、移項、同類項の計算などの初歩的な演算が必要とされる解法も理解させるシステムとなっている。その際、学習者が2次方程式の解を求めるに至るまでの過程を、システムが認識できることが必要とされる。さらに、解を求めるまでに如何なるステップで解に至ったかも認識できるように工夫されている。

本研究では、上述の目的をうけて、システムが学習者に対して適切な診断を提示する点に中心課題がおかれている。

3. システムの構成と基本動作

3.1 システムの構成

上述の目的を達成するために、システムは次のような機能を持つ必要がある。

- 1) 学習者の問題解決過程を認識する機能
- 2) 学習者の誤りを同定し、合理的な診断を行う機能
- 3) 診断に基づいた問題を生成する機能
- 4) 学習対話を円滑に行うためのインタフェースに関する機能

図1にシステムの基本構成図を示す。

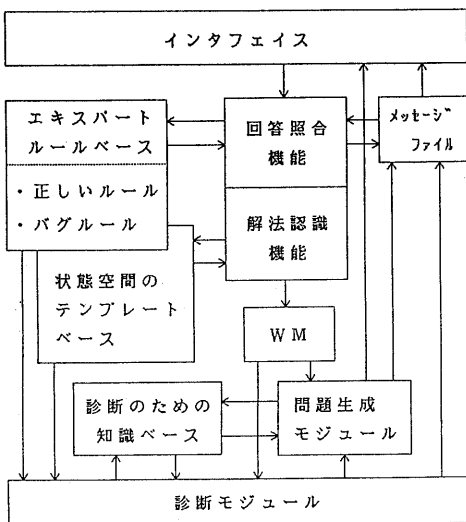


図1 システムの構成

各モジュールは、上述の機能を実現するために、各々独立してその機能を果たす。インタフェースは、ウィンドウの制御や、学習者へのメッセージの提示などを行う。

問題生成モジュールでは、学習者に対して問題を生成する他に、診断結果に基づいた練習問題も生成する。

回答認識モジュールは、学習者の入力をエキスパートルールベースに問い合わせながら、学習者の解法過程を認識する。また学習者の解法過程において、学習者の解法のステップ(状態空間と呼ぶ)も、状態空間のテンプレートベースに問い合わせながら認識する。

診断モジュールは、前述の照合結果に基づいて、エキスパートルールベース、および診断のための知識ベースに問い合わせながら、学習者の誤りを同定する。

メッセージファイルは、回答認識モジュールや、問題生成モジュールの結果を受けて、適当なメッセージをインタフェースに返す。

3.2 基本動作

図2にシステムの基本動作を示す。

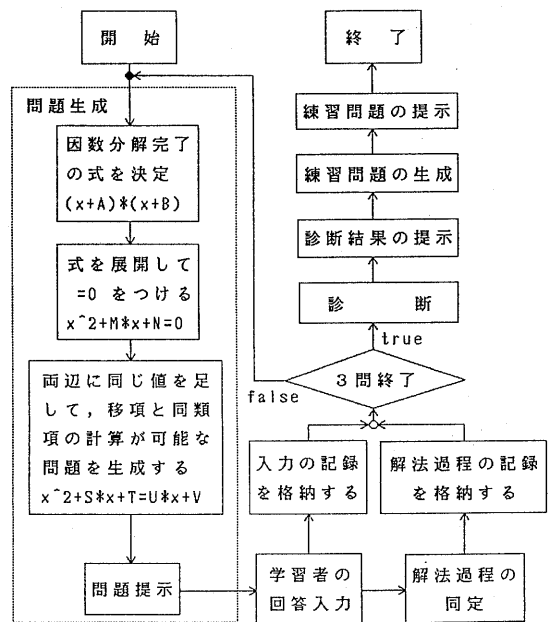


図2 システムの基本動作

- ① システムが因数分解が可能な問題を生成する。
- ・ 因数分解が完了した式 $((x+A)*(x+B))$ を生成する。ただし式の中の A, B は変数で、システム内部で乱数を発生させて、それぞれの変数に発生させた乱数の値を代入する。
 - ・ その式を展開し、それにより生成された式に $(=0)$ をつけて因数分解可能な2次方程式 $(x^2+M*x+N=0)$ を生成する。
 - ・ 解法において移項や同類項の計算を行う式 $(x^2+S*x+T=U*x+V)$ などを生成する。ただし式のパターンは、あらかじめシステム内部で持つ7通りのパターンから、システムがランダムに選択し決定する。具体的には、前述の式 $(x^2+M*x+N=0)$ の両辺に同じ値を足すことにより生成する。
- ② ①で生成された式を、問題として学習者に提示する。
- ③ 学習者が回答を入力する。
- ④ 学習者の回答を、回答認識機能を用いて、エキスパートルールベースを参照しながら認識する。
- ⑤ 学習者の入力が途中段階である場合には、システムはさらに学習者への入力を促す。ただし、システムは各段階において、学習者に対して入力の継続の有無を問う。そこで終了を選択した場合には、その問題に対する学習が終了する。
- ⑥ 学習者が方程式の解を求めるまでに至った場合に、システムは学習者に対して THANK YOU というメッセージを提示する。
- ⑦ システムが学習者に対して、学習者が問題の解法過程において、利用したルールの系列および、個々のルールの説明を提示し、中性的なフィードバックを与える。
- ⑧ ⑦で提示したルールの系列は、解法過程の記録としてワーキングメモリーに格納される。また、学習者の解法における状態空間の系列も同様にして格納される。
- ⑨ ①～⑧のステップをさらに2回行う。ただし問題生成においては、類似問題を生成するようにする。これにより、ルール系列、および解法における状態空間の系列が3つ生成される。
- ⑩ ①～⑧のステップで生成された3つのルール系列および、状態空間の系列を用いて診断を

行う。診断は、解法の冗長性、飛躍性、および順序性の3つの性質において、診断のための知識ベースを参照しながら行う。

- ⑪ この診断結果をもとに、適切なメッセージをメッセージファイルより選択し、学習者に提示する。

- ⑫ 診断結果および、診断のための知識ベースを参照しながら、練習問題を問題生成モジュールで生成し、学習者に提示する。

これでモデリングのプロセスが終了する。

4. 各モジュールの構成

システムは、次のような機能を有する。以下にそれを述べる。

4. 1 インタフェイス

システムは、学習者の回答を正しく入力できるようにする必要がある。そこで、学習者のタイプミスを認識し、それに対して適切なメッセージを表示し、学習者に再入力を促す機能を用意した。さらに、タイプミスではないが、学習者自身が入力した数式に誤りがあることに気付いた場合には、もう一度入力し直すことができるように、学習者に対して、自分の入力が正しいかどうかをシステムが問い直す。もし誤っている場合は、システムは学習者に対して、もう一度入力するように促す。

表示画面においては、問題を常に表示し、学習者の回答は、学習者が入力するごとに表示される。これにより、学習者がその解法過程および、次の回答を容易に認識することができる。また、学習者が学習の中断を要求するときには、リターンキーを押すことで学習の継続、'e'を押すことにより学習の中断が可能となる。

4. 2 エキスパートルールベース

システムは数式を変形するために、ルールを適用する。ここでいうルールは、移項や同類項の計算や因数分解を行うためのルールである。また、ルールは、正しいルールだけでなく、バグルールも含む。ただしルール群は、その性質

に基づいてクラスター化されている。例えば移項に関するルールならば、正しいルールもバグルールも ik という記号がそのルールにつけられている。

ルールの一部を図3に示す。

```

正しいルール
r_rule(ik, X, Y, ikb1):-ikb1(X, Y). /* 左辺から右辺への移項 */
r_rule(rd, X, Y, rdb1):-kdb1(X, Y). /* 左辺での同類項の計算 */
r_rule(in1, X, Y, inb1):-in1(X, Y). /* 定数項の積分解 */
r_rule(in2, X, Y, inb2):-in2(X, Y). /* 因数分解の完成 */
r_rule(an, X, Y, anb1):-answer1(X, Y). /* 因数の抽出 */
r_rule(an, X, Y, anb2):-answer2(X, Y). /* 1次式からの求解 */

バグルール
r_rule(ik, X, Y, ikb1):-ikb1(X, Y). /* 左辺から右辺への移項での誤り */
r_rule(rd, X, Y, rdb1):-kdb1(X, Y). /* 左辺での同類項の計算での誤り */
r_rule(in1, X, Y, inb1):-in1_b1(X, Y). /* 定数項の積分解での誤り */
r_rule(in2, X, Y, inb2):-in2_b(X, Y). /* 因数分解完成における符号の誤り */
r_rule(an, X, Y, anb1):-answer1(X, Y). /* 求解における符号の誤り */

(注) 頭部: 第1引数 クラスター化のための記号
      第2引数 ルール適用前の式
      第3引数 ルール適用後の式
      第4引数 ルール番号
本体: ルールの実行部
/* */ はルールの説明

```

図3 エキスパートルールの一部

4.3 回答認識機能

学習者の解法過程を認識するために、対応するエキスパートルールを適用して、ルール系列の生成を行う。

その方法を次に示す。

- ① 学習者の1つ前の入力(状態空間 $i-1$)と、現在の入力(状態空間 i)との差を評価する。すなわちその差をみとめるための正しい適用ルール、そのルールの順序性および、回数を認識する。
- ② エキスパートルールベースを参照して、対応するルールを選択し、学習者の1つ前の入力(状態空間 $i-1$)にルールを適用させる。
- ③ 適用させた結果により得られた式と、現在入力された式とのマッチングをとる。
- ④ マッチングがとれれば終了。さもなければ、②に戻り、別のルールを適用する。ただし、ルールの適用回数は同じ。

その基本動作を図4に示す。

マッチングについては、式の等価性を評価し得る。学習者がエキスパートルールベースに存在しない誤りを犯した場合には、同定不能ということでシステムは、nil を返す。また学習者が、まだ数式変形が可能な状態の時に学習を終了した場合には、学習者が不完全な理解状態であるという判断のもとで、システムは enil かあるいは knil を返す。enil は、学習者が因数分解を完了させていない状態に対しての値である。また knil は、因数分解は完了しているものの解の展開に至っていない場合の値である。

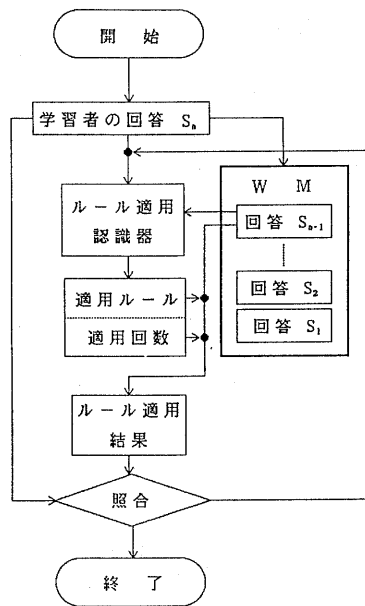


図4 回答認識機能の基本動作

システムは、ある状態空間から次の状態空間に移行する際に、適用されたルールの系列をワーキングメモリーに格納する。ただし1回の状態空間の移行に対して、1つのルール系列をワーキングメモリーに格納するために、ワーキングメモリー内にはいくつかのルール系列が格納されることになる。例えば、状態空間の移行が4回行われた場合には、4つのルール系列が格納されることになる。従って最後に、問題が終

了した時点で、1つのルール系列にまとめる必要がある。このようにして照合がとれたルールの部分系列を1つのルールの系列にまとめ、学習者のプロダクションルール、すなわち理解のモデルとして学習者に提示する。

さらにシステムは、現在の状態空間の1つ前の状態空間が、如何なる状態であるかを、状態空間のテンプレートベースとのマッチングをとることにより認識する。次にそれらの状態空間のテンプレートを示す。

- テンプレート 1 : 左辺への移行のみ可能
- テンプレート 2 : 左辺への移行と左辺での同類項の計算がともに可能
- テンプレート 3 : 左辺での同類項の計算のみ可能
- テンプレート 4 : 右辺での同類項の計算が可能
- テンプレート 5 : 式の整理が可能
- テンプレート 6 : 因数分解可能 1
- テンプレート 7 : 因数分解可能 2
- テンプレート 8 : 求解可能 1
- テンプレート 9 : 求解可能 2
- テンプレート 10 : 求解終了

テンプレートとのマッチングがとれれば、ここで適用されたルールのルール番号も添えてワーキングメモリに格納する。また、各問題における学習者の一連の状態空間の系列も、ワーキングメモリに格納する。

最後に、システムの実行例を図5に示す。

4. 4 診断モジュール

診断モジュールでは、回答認識機能によりワーキングメモリに格納された情報を合成し、学習者の誤りの原因を統合的に同定する。

診断方法は、次のとおりである。

- 1) 3問の問題により生成された、各状態空間において適用されたバグルールを、定量的に合成し、学習者に提示する。
- 2) 3問の問題により生成された、学習者の一連の状態空間の系列を用いて、学習者

の思考過程を、飛躍性、冗長性、および順序性という3つの面から評価して、適切なメッセージを学習者に提示する。

- 3) 診断のための知識ベース、および各状態空間におけるバグルールの情報から練習問題を生成し、学習者に提示する。

```

*-----*
*  次の問題を因数分解で解きなさい。ただし、因数分解した式は必ず書きなさい。
*  [ 問題 ]
*  x^2 - 2 * x + 4 = 3 * x - 2
*-----*
> x^2 + x - 6 = 0 ... 右辺側の式が整理された状態空間
それからどうしましたか ?
> (x-3)(x+2) = 0 ... 左辺側の式が整理された状態空間
それからどうしましたか ?
> x = 3 ... 右辺側の式が整理された状態空間
それからどうしましたか ?
> x = 2 ... 右辺側の式が整理された状態空間
THANK YOU
< 適用ルール >
[ ikb1, ikl, rdl, rdb1, in1, inb3, an1, an2, anb ] ... 適用されたルールの系列
ikl : 右辺から左辺への移項
ikb1 : 右辺から左辺への移項での符号の取り
rdl : 左辺での同類項の計算
rdb1 : 左辺での同類項の計算における符号の取り
in1 : 定数項の因数分解
inb3 : 因数分解完成における符号の取り
an1 : 因数の抽出
an2 : 1次式からの求解
anb : 求解における符号の取り
(注) アンダーライン 学習者の入力
      裏付け 説明文
  
```

図5 システムの実行例

1)については、学習者の一連の状態空間すべてについて、各々の状態空間において、バグルールが何回使用されたかを定量的に求める。ここでいうバグルールは、nil, enil, knil など含まれる。

2)については、まず学習者に提示された問題が解法過程において、どのような順序性を有する状態空間の系列が必要なのかをシステムが認識する。さらに上述の状態空間の系列と、ワーキングメモリに格納されている学習者の状態空間の系列とを比較し、前述2)の3つの側面から診断に必要な情報を推論し、それをワーキングメモリに格納する。

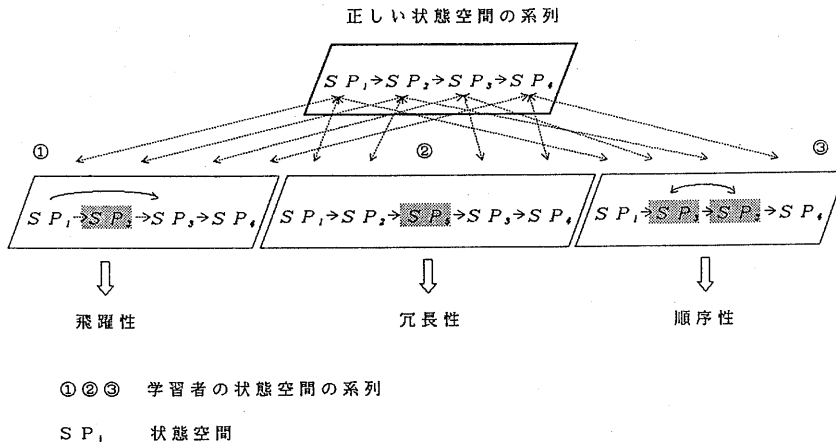


図6 状態空間の関係

① 飛躍性について

その問題が持つ、正しい順序性を備えた、必要不可欠な状態空間の系列と、学習者の一連の状態空間の系列とを比較し、欠けている状態空間を抽出する。その場合に、3つの問題における状態空間の系列すべてに共通して欠けている状態空間だけではなく、少なくとも2つの状態空間の系列に欠けている状態空間も抽出する。また状態空間の系列の中に、同じ状態空間が2回以上存在することもありうる。その場合には、診断のための知識ベースを参照し、その問題において同じ状態空間がいくつ存在するかを計算し、それをもとに、欠けている状態空間を抽出する。

② 冗長性について

飛躍の場合と同様に、その問題がもつ、正しい順序性を備えた、必要不可欠な状態空間の系列と、学習者の一連の状態空間の系列とを比較する。そこで、学習者の一連の状態空間の系列の中に、余分な状態空間が存在した場合には、システムは学習者が冗長な解法を行っているかと判断して、余分な状態空間をワーキングメモリーに格納する。この場合にも飛躍の場合と同様に、すべての状態空間の系列に共通して存在している余分な状態空間のみならず、少なくとも2つの状態空間の系列に存在しているものを抽出する。

③ 順序性

本システムは移項と同類項の計算に限定したものである。システムは移項と同類項の計算に関して、まず移項を行ってから同類項の計算を行うという考え(信念)に基づいて、順序性に関する診断を行う(これは、中学校における数学的指導方法からみた場合に、標準的なものである)。順序性に関して、学習者の一連の状態空間の系列を参照する。システムは移項と同類項の計算の順序に関して、各問題において学習者が誤っていると予想される知識を、最初に診断のための知識ベースの中に、状態空間の系列として有している。その知識と、前述した学習者の状態空間の系列とを比較して、順序性に関して誤っているかどうかを判断し、その結果を再びワーキングメモリーに格納する。

状態空間の関係を、図6に示す。

以上飛躍性、冗長性、順序性に関する3つの情報をもとに、診断を行う。

飛躍性における診断は、まず飛躍された状態空間の1つ前の状態空間、および1つ後の状態空間を抽出する。そこで抽出された1つ前の状態空間において、学習者が行った誤りを、1)において合成された状態空間における誤りの情報から抽出して、学習者に対して適切なメッセージを提示する。その場合に、学習者がどの状態空間からどの状態空間に移るときに飛躍をしたのかということ、先ほど抽出した前後の状態

空間から認識して、学習者に提示する。

冗長性の診断は飛躍の場合と同様に、ワーキングメモリーを参照して、余分な状態空間の前後の状態空間を抽出する。そこで余分な状態空間の1つ前の状態空間における誤りを、1)で合成された状態空間における誤りを参照して、学習者に対して適切なメッセージを提示する。冗長性についても飛躍と同様に、前後の状態空間の情報も学習者に提示する。

順序性の診断に関しては、移項と同類項の計算において、学習者が飛躍した考えを行った場合には、移項と同類項の計算に関するルールが変換可能なために診断は行えない。すなわち、学習者が移項と同類項の計算を同時に行った場合には、移項をすべて行ってから同類項の計算を行ったのか、移項と同類項の計算を交互に行ったのかを認識することが不可能だからである。また、システムは、学習者が冗長な考えを行った場合には、順序性に関する診断より、冗長性に関する診断を優先する。以上の条件のもとに、システムは順序性に関する診断を行う。そこでシステムが、学習者が誤った順序で解法したと評価した場合には、同類項の計算より、移項を優先して行うように、適切なメッセージを学習者に対して提示する。診断の基本動作を図7に示す。

最後に3)に関して述べる。システムは、1)で生成された状態空間におけるバグルールの情報と、診断のための知識ベースが有する、バグルールに関する知識を参照して練習問題を生成する。システムはまず、各々の状態空間において、1番多用されたバグルールを抽出する。そこで抽出された各バグルールが、どのような問題において使用されるかを、診断のための知識ベースが有する問題生成のための知識を参照することにより認識する。

問題生成のための知識には、因数分解を行うことができる数式($x^2+Ax+B=0$)を生成するための知識と、今述べた数式から移項や同類項の計算を含んだ数式($x^2+2Ax+B=Cx+D$ など)を生成するための知識が、各バグルールにおける問題生成のための知識として格納されている。

そこで、該当するバグルールに対する、上述の2つの知識を参照して、問題を生成する。ただし、前述した、状態空間から抽出されたバグ

ルールが2つ以上存在した場合には、抽出されたすべてのバグルールが使用され得る問題を生成できるように、前述の2つの知識を各々1つに同定し、その情報に基づいて、問題生成モジュールにおいて、適切な練習問題を生成し学習者に提示する。

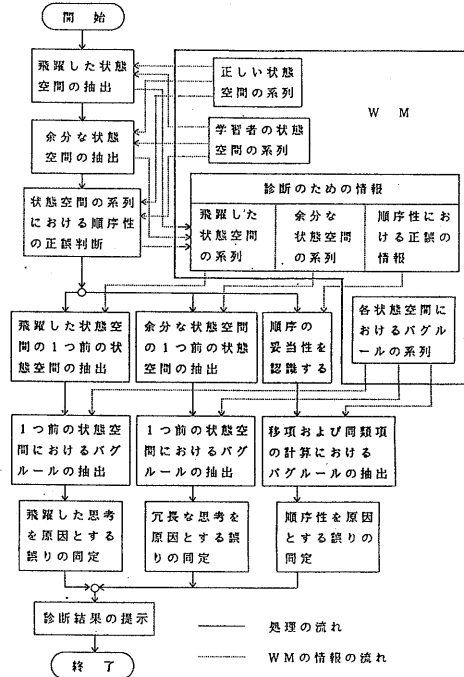


図7 診断の基本動作

5. 今後の課題

問題における学習者の解法過程には、様々なものがある。それに対して本システムのエキスパートは、解法が非常に限定されている。従って、学習者が特殊な解法を行った場合に、システムが認識できずに、nilを返してしまうことがありうる。そこで、学習者の誤りを認知心理学的に分析して、バグルールおよび、状態空間の内容を検討し、エキスパートルールベースおよび、状態空間のテンプレートベースを充実させる必要がある。

また、本システムにおける診断は、なお定量的である。従って、真の学習者の誤りを同定できないことがある。そこで学習者の解法過程、および状態空間の系列を利用して、さらに高次

な推論が行えるエンジンをシステムにつけることが今後の課題である。

6. 参考文献

- [1] 岡本敏雄：知的 C A I のための教授
世界知識の表現とその推論方法，
電子情報通信学会論文誌，VOL.J70-D，
No.12，pp.2658-2667，(1987，Dec.)
- [2] 岡本敏雄，松田昇：幾何論証の学習世界
における知的 C A I の構成について，情
報処理学会論文誌，VOL.29，No.3，pp.3
11-324 (1988，Mar.)
- [3] Polson.M.C and Richardson.J.J: Foun-
dation of INTELLIGENT TUTORING SYS-
TEMS, LAWRENCE ERLBAUM ASSOCIATES
PUBLISHERS (1988)
- [4] Sleeman.D: Assessing aspects of com-
petence in basic algebra, Sleeman.D
and Brown.J.S, Intelligent Tutoring
system, pp185-199, ACADEMIC PRESS
(1982)