

# 直接解決可能性に基づく一般化：DSBG

## —操作可能な SOLVABLE 概念の定義づけ—

### DSBG : Direct Solvability-Based Generalization

#### —Defining operational SOLVABLE concept—

山田 誠二\*<sup>1</sup> 辻 三郎\*<sup>1</sup> 安部 憲広\*<sup>2</sup>  
Seiji Yamada Saburo Tsuji Norihiro Abe

- \*1 大阪大学基礎工学部制御工学科  
Dept. of Control Eng., Faculty of Engineering Science, Osaka University, Toyonaka 560, Japan.
- \*2 大阪大学産業科学研究所  
The Institute of Scientific and Industrial Research, Osaka University, Osaka 567, Japan.

1988年4月21日 受理

**Keywords:** machine learning, explanation-based generalization, operability, direct solvability, macro-operator.

#### Summary

We have developed PiL system : a learning system that can acquire strategy knowledge from training examples with EBG (Explanation-Based Generalization) method. EBG, however, is not sufficient in generalization, therefore many redundant training examples were needed for learning. The under-generalization results from the generalization without SOLVABLE concept proposed by Mitchell *et al.*, but the non-operability of the concept prevents from its usage. Thus we propose how to define operational SOLVABLE concept and a new generalization method DSBG (Direct Solvability-Based Generalization), an EBG method extended with DS (Directly Solvable) concept that is an operational SOLVABLE concept. With DSBG, PiL generalizes the examples more sufficiently than EBG and the significant decrease in the amount of the training examples is possible.

## 1. はじめに

近年、「説明に基づく一般化 (EBG : Explanation-Based Generalization) あるいは学習 (EBL : Explanation-Based Learning)」が機械学習におけるトピックスとなっており、さまざまな研究が活発に行われている<sup>(1)-(8)</sup>。従来の帰納的一般化において問題であった全体的な概念、ハイアラーキによる無根拠な一般化である帰納的飛躍 (inductive leap)<sup>(1)</sup> や膨大な訓練例が必要なことによる学習効率の悪さなどを克服するものとして、事前に与えられた正しい領域知識から演繹的な一般化を行う EBG (あるいは特殊化も行う EBL) が提案され、総括的な論文<sup>(1)(2)</sup> も発表されている。しかし、なぜか国内での研究例<sup>(9)(10)</sup> はわずかであり、まだこれからの感がある。

筆者 (山田) 自身もすでに、EBG を用いた問題解決

における戦略知識学習システム : PiL<sup>(10)</sup> を構築してきたが、その過程で従来の EBG の問題が生じた。それは、従来の EBG の不十分な一般化 (under-generalization) により、教師にとって冗長に思われる訓練例を数多く教示しなければならないことである。筆者はこの問題点の主要な原因は、Mitchell らの提案した解決可能な問題状態の内包である SOLVABLE 概念<sup>(11)(12)</sup> が操作可能 (operational)<sup>(1)(2)(14)</sup> ではない<sup>(11)-(13)</sup> ために、EBG が SOLVABLE 概念を用いた一般化、つまり説明木の途中からの一般化ができないことであると考え、本論文において SOLVABLE 概念の操作可能なサブセットである DS (Directly Solvable) 概念を定義し、DS 概念を用いて EBG を拡張した「直接解決可能性に基づく一般化 (DSBG : Direct Solvability-Based Generalization)<sup>(15)(16)</sup> を提案する。さらに、DSBG を実際に PiL システムに適用して 1 次・2 次・分数・指数・対数方程式の解

法学習の実験を行い、従来のEBGを用いた実験結果との比較・検討によりDSBGの有効性を検証する。

まず最初に、PiLシステムの概要およびEBGの問題点について述べ、次にDSBGアルゴリズム、学習実験の評価基準を説明し、最後に実験結果と検討を示す。

## 2. PiLシステムの概要

PiLシステム<sup>(10)</sup>は、問題解決のための基本的な書換えルール(基本オペレータ)と問題の解決過程を表す解法例を与えられ、それらから基本オペレータをどういふ場合に適用すればよいかという戦略知識を獲得する学習システムである。PiLは、プロダクションシステムにより問題解決を行う問題解決モジュール、得られた解法例を一般化し、そこからマクロオペレータなどを抽出する知識管理モジュールおよび知識ベースから構成されている。知識ベースは階層構造になっており、最下層に基本オペレータが、上層に獲得された戦略知識のオペレータが置かれ、上層から優先的に適用されていく。また、将来的にはPiLシステムは汎用学習システムを目指しているが、現在のところ方程式の解法学習における応用を行っている。以下、後述するDSBGに関連するPiLのEBGとマクロオペレータについて簡潔に述べる。

### 2.1 PiLにおけるEBG

PiLは、問題の初期状態から解に至るまでの解法過程をPiL自身が持っているオペレータで埋めた解法例をEBGにより一般化していく。詳細は文献(10)に譲り、ここでは簡単な例としてFig. 1に1次方程式の解法例(A)とそれがEBGにより一般化されていく過程(B)を示す。図中のルールは適用されたルールを表す。

図中で、R0nは任意の実数、A, B, Cは任意の整式、NAn, NBは0以外の任意の整式、LS, RSは左辺・

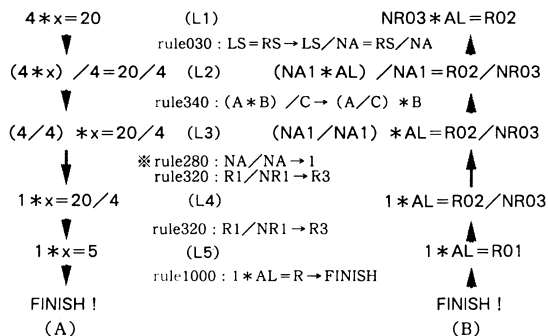


Fig. 1 EBG in PiL.

右辺, NR0nは0以外の任意の実数, ALは任意のアルファベット(変数)を表す(以降, オペレータの表記はこの表現に従う)。一般化の伝播は終了条件から始まる。まず、終了条件rule 1000の条件部の左辺は1と任意のアルファベットの積、右辺は任意の実数でよいので、Fig. 1 (A)のL5の左辺の“x”は任意の変数ALに、右辺の“5”は任意の実数R01に一般化され、Fig. 1 (B)のL5のようになる。次にrule 320により、(A) L4の右辺の“20”と“4”はR02, NR03に一般化される。さらに、L3ではrule 280とrule 320が競合しているが、ある基準でrule 280が最適経路として選ばれ<sup>(10)</sup>、rule 280により左辺の二つの“4”が任意の0でない整式NA1に一般化される。あとは同様にL1まで一般化が行われていく。

次に実際の一般化手順を述べる。解法例が得られるとまず、解法シーケンスの各行が変数化される。「変数化」とは、オペレータの条件部の構造を残して等式を最大限に一般化することである(Fig. 2)。そして、

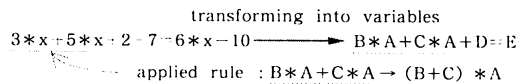


Fig. 2 Transformation into variables.

その変数化された解法シーケンスの各行を終了条件から単一化(unification)していき、さらに関係する拘束条件を付け加えることにより一般化が行われる。ここで重要なことは、PiLのEBGは説明木の根である終了条件ルールから一般化が後方連鎖することである。このため後述するような不十分な一般化が行われることになり、それを改善するのがDSBGである。

### 2.2 マクロオペレータの抽出

問題解決のパフォーマンス効率化のため、PiLは解法例中の特定のオペレータシーケンスをマクロオペレータとして抽出する。しかしこのとき、マクロオペレータの候補は多数あるので、それらの中から役に立つものだけを選択しなければならない。PiLは、「人間は、オペレータ間に閉じた強い因果関係があるときにそのオペレータシーケンスをマクロオペレータとして取り込む」という仮定に基づいた完全因果性<sup>(10)</sup>という基準でマクロオペレータの抽出を行う。抽出されたマクロオペレータのうち、ワンステップで問題を

- M-1 :  $R1 * AL + R2 = R3 * AL \rightarrow 1 * AL = R4$
- M-2 :  $R1 * AL = R2 * AL + R3 \rightarrow 1 * AL = R4$
- M-3 :  $R1 * AL = R2 \rightarrow 1 * AL = R3$
- M-4 :  $R1 * AL * AL + R2 * AL + R3 = R4 \rightarrow 1 * AL = R5$
- M-5 :  $R1 * AL * AL + R2 = R3 * AL + R4 \rightarrow 1 * AL = R5$

Fig. 3 Generated DS macro operator.

解いてしまうマクロオペレータを直接解決可能なマクロオペレータ (DS マクロオペレータ) と呼ぶ。なお、完全因果性により有効なマクロオペレータが抽出されることが、1 次方程式での実験<sup>(10)</sup>、さらに今回の各種方程式での実験 (本論文 6 節) で確かめられている。抽出された DS マクロオペレータの例を Fig. 3 に示す。

$$\begin{aligned} \frac{3*x}{3*x+2} &= 5 & \frac{3*x+4}{3*x+2} &= 5 \\ \frac{3*x^2+6*x+1}{3*x+2} &= 5 & \frac{3*x^2+6*x}{3*x+2} &= 5 \\ \frac{3*x^2+1}{3*x+2} &= 5 & \frac{3*x^2}{3*x+2} &= 5 \end{aligned}$$

Fig. 5 Redundant examples.

### 3. PiL における EBG の問題点

一般に EBG の問題点として、正しく完全な領域知識 (domain theory) と訓練例が与えられなければならない<sup>(11)</sup> ということがあるが、実際に PiL システムに学習をさせていく過程で、もう一つの問題が生じた。それは一般化が十分にでないために、学習効率が悪いことである。EBG は従来の帰納的一般化に比べると、一つの訓練例からの一般化が可能であり非常に学習効率がよい。しかし、実際 PiL システムへの教示で、1 次・2 次方程式からさらに分数方程式、指数・対数方程式へとその解法が複雑になるに従って、教師が教示しなければならない解法例の数も増え、かつ、それらの中には教師から見ると冗長なものかなり多く含まれていることがわかった。

冗長な教示の具体例を Fig. 4 で説明する。図中で (A) は解法例を、(B) は EBG による一般化伝播の過程を簡略化して示している。ここでは 1 次の分数方程式を教示しているが、この時点で PiL は 1 次・2 次方程式の解法をすでに学習済みで、すべて解けるとする。この解法例から完全因果性により Fig. 4 の解法シーケンス全体が一つの DS マクロオペレータ: DSM\_1 として抽出される。ここで (A) L1 の左辺の分子である '4' に注目してほしい。この '4' は (B) のように EBG による一般化が終了状態から伝播されることにより、(B) L1 に示すように任意の実数 R4 に一般化される。よってこの「両辺に左辺の分母と同じ整式を掛けて解く」というマクロオペレータ: DSM\_1 は、この一つの解法例からだけでは、左辺の分子が実数でなければ適用されない。しかし、そのほかにもこのマ

クロオペレータが適用されることが望ましい Fig. 5 のような問題状態、つまり、左辺の分子が 2 次以下の任意の整式である場合が容易に考えられる。しかし、PiL は Fig. 4 の教示だけではこれらの問題を解くことができず、さらに Fig. 5 のような問題それぞれについて解法例を教示する必要があった。このことは終了状態からの一般化が不十分であることを表している。このように冗長に思える解法シーケンスを多数教示することは教師にとって退屈かつ不快であり、Fig. 4 の一例を与えれば Fig. 5 の問題を独力で解けるように一般化することが望まれる。それを実現するのが DSBG である。

### 4. 直接解決可能性に基づく一般化: DSBG

従来の PiL で得られる DS マクロオペレータは、必ず解に到達できる場合にだけ適用されていた。しかし、人間が過去に解いたことのない問題に出会ったとき、解にたどり着く確信がなくてもとにかく知っている方法を適用して途中まで解き、その結果が解けそうなものかどうかをみるという柔軟な問題解決を自然に行っている。このような戦略の効率が保証されるには、途中まで解いた結果が解決可能かどうかを容易に検証できなければならない。つまり、解決可能という概念が操作可能 (operational)<sup>(11)(12)</sup> である必要がある。操作可能性 (operationality) とは EBG における重要な概念で、ある具体例がある概念に属するかどうかを容易に検証できるとき、その概念は操作可能であるという。なお、操作可能性については最近 Keller<sup>(14)</sup> によってより厳密な定義が提案されている。

以上のような考えのもとに、EBG と DS マクロ

$\frac{4}{3*x+2} = 5 \quad (L1)$ $\frac{4}{3*x+2} * (3*x+2) = 5 * (3*x+2) \quad (L2)$ $4 = 15*x + 10 \quad (L3)$ $-15*x = 6 \quad (L4)$ $1*x = -2/5 \quad (L5)$ <p style="text-align: center;">(A)</p>	$\frac{R4}{(R7*AL+R8)} = R9$ $\frac{R4}{(R7*AL+R8)} * (R7*AL+R8) = R9 * (R7*AL+R8)$ $R4 = R5*AL + R6$ $R2*x = R3$ $1*AL = R1$ <p style="text-align: center;">(B)</p>
--	--

Fig. 4 The teaching to solve a fractional equation.

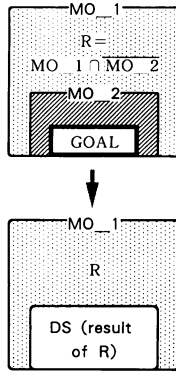


Fig. 6 DSBG.

ペレータを用いてより十分な一般化を行う直接解決可能性に基づく一般化 (DSBG) を提案する。

まず、直接解決可能性について述べる。直接解決可能な問題状態とは、それまでに蓄積された DS マクロオペレータだけを用いて解決できる問題状態であると定義し、そのような問題状態の集合を DS 概念と呼ぶ。直接解決可能性を検証する述語 DS は、引数である問題状態を今まで蓄積された DS マクロオペレータだけの適用で実際に解いてゆき、解決できれば真を返し、適用される DS マクロオペレータがなくなったときに偽を返す。この DS マクロだけによる探索は、すべてのオペレータを用いた問題解決よりも遙かに探索空間が小さく、よって相対的に検証が容易である。つまり、DS 概念は人間が一目見て解けると判断できる問題状態の集合に対応していると考えられる。また、DS 概念は学習の過程で新たな DS マクロオペレータが得られるたびに更新される。

次に、Fig. 6 で DSBG アルゴリズムについて説明する。解法例が与えられるたびに、まず完全因果性に基づきマクロオペレータが抽出される。そして得られた DS マクロオペレータ：MO\_1 が他の DS マクロオペレータ：MO\_2 を包含しているときに MO\_1 が DSBG の対象となる。DSBG は、解法シーケンスから MO\_2 を切り離れた残りの領域  $R = MO_1 \cap MO_2$  内のみで EBG による一般化を行う。このとき、2・1 で述べた一般化伝播の最適経路選択<sup>(10)</sup>を改めて行う。これは DSBG では終了状態から EBG により一般化した場合とは異なった一般化経路が選択される可能性があるからである。また、最大限に一般化するために、包含される DS マクロが複数個ある場合は、そのうち最大のものを取り除く。

このような一般化を行い、実際の問題解決時には解法シーケンス R を通過して得られた結果を述語 DS でチェックし、それが真なら問題は解決されたことにな

るし、偽ならその DS マクロの適用は失敗し次のオペレータを探す。一般化伝播の際に、MO\_2 の拘束条件は伝播されないので、DSBG は MO\_1 の条件部を EBG の場合よりもさらに一般化することが可能であり、さらに述語 DS によるチェックをしているので過度に一般化されたオペレータを無駄に適用して、いたずらに探索空間を広げることを避けることができる。人間の問題解決との比較では、R の部分 (DSBG によって得られる DS マクロ) で問題を解くことがとにかく途中まで問題を解いてみることにあたり、述語 DS によるチェックが、その結果が過去に解けた問題かどうかを判断することに対応する。

以上により、従来の PiL の EBG よりもさらに十分な一般化を行い、そして実際の問題解決時には問題を途中まで解いて、それが現在の知識で解けるものかどうかを考えてみるという戦略が実現される。

DSBG の具体例として、Fig. 4 を DSBG で一般化していく過程を Fig. 7 に示す。図中で (A) は解法例を表し、(B) は各行が DSBG により一般化された結果を表している。(A) で L1 からオペレータが順次適用されて L8 において DS マクロオペレータ：MO\_2 が適用され、目標状態に直接的に到達する。つまり、L8 において直接解決可能になる。ここでは、完全因果性に基づき L1~Ln が一つの DS マクロオペレータ：MO\_1 となり、それが L8~Ln の DS マクロオペレータ：MO\_2 を包含しているので、MO\_1 が DSBG の対象になる。そこで MO\_2 に含まれる L8~Ln までのシーケンスが削られ、残りの部分でまず各行が前述のように変数化され、EBG による一般化が行われる。なお、この MO\_2 は文献 (10) で完全因果性により実際に獲得された DS マクロオペレータである。この一般化で L5 から L4 に伝播する際、rule 250 と rule 310 の二つの伝播経路が考えられ、PiL は改めて最適経路の選択を行う。ここでは (B) の L4 の変数 A は任意の整式でいいので rule 250 が選択される。詳細は省くが目標状態から EBG により一般化した場合は、逆に rule 310 の方が最適経路として選択される。以下同様に L1 まで一般化の伝播が行われた結果が Fig. 7 の (B) である。ここで注意すべき点は、前述のように PiL の EBG はまず式をできるだけ一般化 (変数化) した後、特殊化するということである。(B) の L1 を見るとわかるように、Fig. 4 の EBG では任意の実数 R4 に一般化されていた左辺の分子が任意の整式 A に一般化されている。これは変数化によって任意の整式に一般化された A を、EBG では任意の実数に特殊化した拘束条件が、取り除かれた MO\_2 内に含まれて

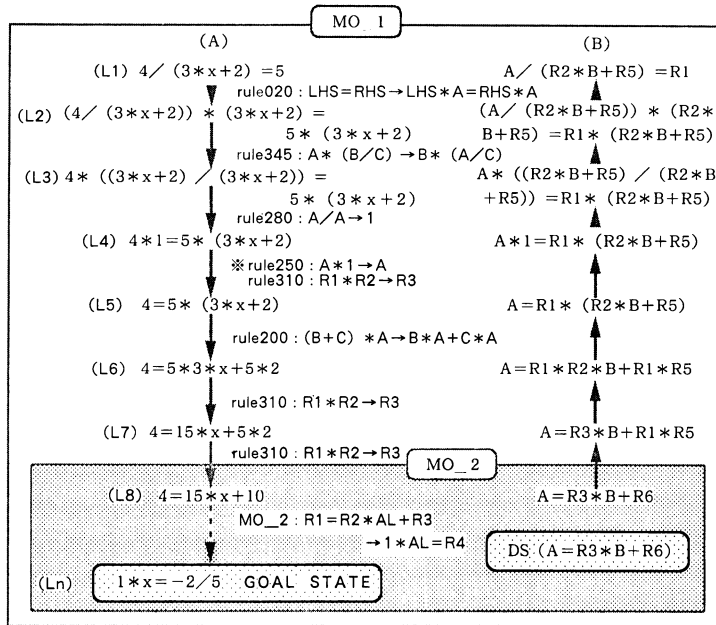


Fig.7 Generalization with DSBG.

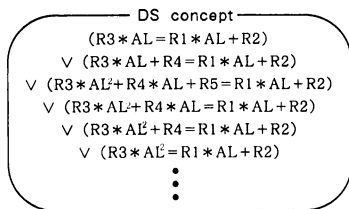


Fig. 8 DS concept.

いるので伝播しなかったためにAのまま残ったからである。

さて、このように DSBG により一般化された DS マクロオペレータが得られた後、実際の問題解決時にはこの任意の整式Aは L8 で述語 DS によりチェックされる。この時点で1次・2次方程式は、学習済みなのでDSマクロの条件部の集合であるDS概念はFig.8のような状態である。よって、AはFig.8の要素の等式の左辺であればよく、そのときMO\_1が適用されることになる。つまり、Fig.4の一例を教示することにより、Fig.5のすべての問題が解決可能になり、教示しなければならぬ訓練例が減少し学習の効率化が実現される。これは分数方程式の場合に限ったことではなく、他の指数・対数方程式などにおいてもDSBGにより学習効率が向上すると考えられる。

次に、DSBGにより得られたDSマクロオペレータを知識ベースのどの階層に入れるかであるが、これは文献(10)の知識の階層を改良して、Fig.9のように絶対オペレータとEBGのマクロオペレータ(DSマクロを含む)の階層の下に新しい階層を作ってそこに

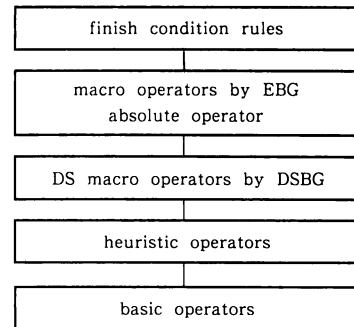


Fig. 9 Hierarchy of knowledge base.

入れる。これにより、過去に解いたことのあるパターン問題はEBGのDSマクロで迅速に解き、未知の問題に対しては下層のDSBGのDSマクロで試行錯誤しながら解く、という人間に近い解き方を実現できる。

### 5. 学習能力の評価基準の選定

本節では、DSBGとEBGの学習能力を比較するための評価基準の選定を行う。ここで注意すべき点は、一般化プロセスの改善により学習システムの能力の一部が向上しても、それに対してシステムの問題解決パフォーマンスが低下し、システム全体の能力の改善になっていない場合が起こり得ることである。よって、できる限りシステム全体を総括的かつ本質的に評価できるような基準を選定することが重要である。一般に、問題解決における例題からの学習手法の能力を評価す

る基準として、以下のものが考えられる。

- (1) システムが、あるレベルの学習状態に達するために必要な訓練例の教示にかかるコスト。
- (2) 学習後の問題解決のパフォーマンスが学習前よりどれほど向上したか。
- (3) 学習プロセス自体にどれほどのコストがかかるか。

この三つの評価基準のうち、ここでは、(1)と(2)のみに着目する。なぜなら、(3)は学習プロセスをいかにうまくインプリメントするかに依存するので、本質的な意味が希薄だからである。次に、(1)、(2)について PiL に照らし合わせながら、さらに具体的に考えてみる。

まず、(1)についてであるが、従来はこの訓練例教示のコストを定量的に考慮することはほとんどなかった。しかし、PiLのようなマン・マシン・インタラクティブな学習システムにおいては、この(1)のコストは非常に重要である。そこで PiL では(1)のある学習レベルに達するまでに、PiL 自身が解けなくて教師が教示しなければならない訓練例(必須訓練例と呼ぶ)の個数で評価する。当然、この必須訓練例の個数が少ない一般化手法の方が優れていることになる。ここで、PiL が独力で問題を解けない状態とは、ワーキングメモリ中に任意の整式(変数)を含むファクトができた場合(このとき展開は爆発する)、または延べのファクト数が100個以上になっても解けない場合であるとする。

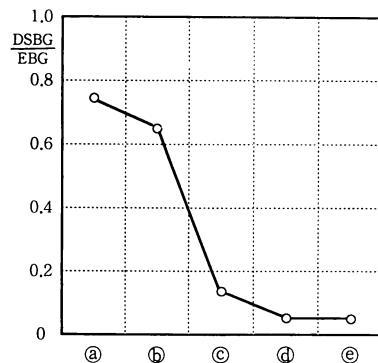
次に、(2)の問題解決のパフォーマンス向上であるが、これは学習後にどれだけ探索空間が縮小するかであ

る。PiL では学習前の探索空間は EBG, DSBG によらず同じであるので、学習後の問題解決時に展開された探索木のノード数、つまり学習後のパフォーマンスにおけるワーキングメモリ中のファクト数を調べる。

### 6. 訓練例教示と実験結果

今回は1次・2次方程式および分数・指数・対数方程式の教示を行い、前節の評価基準で EBG と DSBG を比較した。本実験での訓練例の教示法を以下に述べる。教師はまず中学・高校用の問題集を参考に各種方程式の問題を、教示戦略なしに EBG を用いた PiL, DSBG を用いた PiL それぞれに与えていく。そして PiL が独力での問題解決が不可能なとき、教師が解法例を与える。このような教示法で、1次方程式89問、2次方程式212問、分数方程式35問、指数・対数方程式それぞれ78問を与え、教示した結果、二つの PiL は与えられた問題にすべて独力で解答可能な状態に至った。Fig. 10 に EBG, DSBG それぞれにおける必須訓練例(necessary examples)の個数とその比を表すグラフを、Fig. 11 に具体例として対数方程式での必須訓練例(EBG は一部)を示す。今回の実験における必須訓練例とは、与えたすべての問題を独力で解ける学習状態に達するまでに教師の解法教示が必要な訓練例(方程式)を意味する。また、問題解決のパフォーマンス向上の評価として、ファクト数を調べた

	EBG	DSBG
equations of the first degree ①	8	6
equations of the second degree ②	45	29
fractional equations ③	35	5
logarithmic equations ④	57	3
exponential equations ⑤	57	3



<The numbers of necessary examples>  
 <The ratio of DSBG's necessary examples to EBG's ones>  
**Fig. 10** Experimental results.

DSBG	EBG
$\log(10, 5x) = 2$	$\log(10, 5x) = 2$ $2 = \log(10, 5x - 5)$
$\log(10, 6x+3) = \log(10, 4x-5)$	$\log(5, 3x-4) = 1$ $\log(10, 6x+3) = \log(10, 4x-5)$
$\log(5, 18x) + \log(5, 2x+6) = 3$	$\log(10, 2x^2) = 2$ $\log(10, 5x) = \log(10, 2)$
	$\log(3, 4x^2-1) = 5$ $\log(2, 2x+6) = \log(2, 34)$
	$\log(4, 8x^2+2x) = 2$ $\log(5, 23) = \log(5, 5x-5)$
	$\log(10, 2x^2-3x-10) = 1$ $\log(4, 6x) + 3 = 7$
	$2 = \log(10, 2x)$ $\log(4, 6x+3) + 3 = 7$

**Fig. 11** Necessary examples in logarithmic equations.

$$\begin{array}{l}
 \langle \text{DSBG} \rangle \\
 \text{NA} + \text{R1} = \text{R2} \rightarrow \text{NA} = \text{R3} \\
 \log(\text{A}, \text{B}) = \log(\text{A}, \text{C}) \rightarrow \text{B} = \text{C} \\
 \\
 \langle \text{EBG} \rangle \\
 \text{R1} * \text{AL} + \text{R2} = \text{R3} \rightarrow 1 * \text{AL} = \text{R4} \\
 \text{R1} * \text{AL} * \text{AL} + \text{R2} = \text{R3} \rightarrow \text{A} * \text{AL} = \text{R4} \\
 \log(\text{A}, \text{R1} * \text{AL}) + \text{R2} = \text{R3} \rightarrow 1 * \text{AL} = \text{R4} \\
 \vdots \\
 \log(\text{A}, \text{R1} * \text{AL}) = \log(\text{A}, \text{R2}) \rightarrow 1 * \text{AL} = \text{R3} \\
 \log(\text{A}, \text{R1} * \text{AL} + \text{R2}) = \log(\text{A}, \text{R3} * \text{AL}) \rightarrow 1 * \text{AL} = \text{R4} \\
 \log(\text{A}, \text{R1} * \text{AL} * \text{AL}) = \log(\text{A}, \text{R2} * \text{AL}) \rightarrow 1 * \text{AL} = \text{R3} \\
 \vdots
 \end{array}$$

Fig. 12 DSBG's DS macro operators and EBG's ones corresponding to them.

が、その結果、前述の知識ベースの階層構造の設定により、EBGとDSBGでファクト数は全く同じであり、パフォーマンスの差異はなかった。ちなみに、この実験での2次方程式の解法は完全平方の形にして解く方法のみを教示した。また、1次・2次方程式はほぼすべての問題が解けるレベルまで教示し、その他の方程式では基本的な問題だけを教示した。

## 7. 実験結果の評価

Fig. 10から明らかのように、DSBGではEBGよりも必須訓練例の個数が大幅に減少している。特に、分数・指数・対数方程式などの複雑な方程式でその傾向が顕著になっている。また、ファクト数は変化していないので、問題解決のパフォーマンス効率を損なうことなく、必須訓練例の教示コストを大幅に減少させてシステムの総合的な学習能力向上に成功しているといえる。これにより、DS概念が非常に有効なことが検証された。

以上のような必須訓練例の減少の起因となった、DSBGで得られたDSマクロオペレータとそれに対応するEBGのDSマクロオペレータをFig. 12に示す。この図からDSBGにより移項、logを外すなどの人間がよく使う解法がEBGよりも一般化された状態で獲得されており、これらが学習効率の向上に貢献していることがわかる。

## 8. EBG研究におけるDSBGの位置づけ

本節では、DSBGと他のEBG研究との関連について述べる。まず、MitchellらのLEX2の研究<sup>(1)(12)</sup>でのSOLVABLE概念は文献(1)(11)(13)でも述べられているように、ある訓練例がある概念のインスタンスであるのかどうかを容易に検証できないという意味で操作可能ではなかった。EBGは、「目標概念は操作可能な概念述語で記述されなければならない」という操作性基準(Operationality Criterion)<sup>(1)</sup>に基

づいて説明木を展開するので、LEX2では従来のPiLと同様に解法シーケンスの途中から一般化伝播させることができず、問題が解決された最終状態(説明木の根)から一般化伝播を行っていた。よって、得られた概念は当然必要以上に拘束が強いものとなり、その結果PiLでは冗長な必須訓練例が多数必要となったわけである。Mitchellらも領域に依存した操作可能なSOLVABLE概念(例えば、不定積分においては、あらゆる多項式の積分はsolvableである)を与えることについて述べている<sup>(11)</sup>が、その一般的な定義を与えることはできていない。それに対してDSBGでは、まず「DS概念は、DSマクロオペレータだけを用いた探索により解決可能な問題状態の集合である」と領域独立な定義をする。「実際に探索してみればsolvableである」という検証法は本質的に正しいものであるが、あらゆる種類のオペレータを適用していたのでは探索空間が広がりすぎて操作可能な概念にはならないし、また、その検証過程自体が実際に問題解決を行うことと等価になる。DS概念の定義は、DSマクロオペレータのみを用いて探索を行うことにより、すべてのオペレータを用いた探索よりも相対的に探索空間を縮小させて、訓練例がある概念に含まれるかどうかの検証を容易にすること、つまりDS概念を操作可能にすることを意図している。そして、後は対象分野における実際の学習過程で領域独立な基準(完全因果性<sup>(10)</sup>)に基づきDSマクロオペレータを獲得することにより、操作可能なDS概念を自動的に生成することが可能である。ただし、DS概念は、解決可能な問題状態すべてを表すSOLVABLE概念と一致するものではなく、SOLVABLE概念の操作可能な部分集合であるといえる。

## 9. DSBGの問題点

Fig. 7の一般化から明らかのように、DSBGを用いても左辺の分母と右辺を任意の整式にまで一般化することはできない。これは、DSマクロオペレータを

取り除いた残りの解法シーケンス (R) の中に左辺の分母と右辺を拘束する条件部をもったルール (Fig. 7 の rule 200, rule 310) が含まれているからである。このことは DSBG でさえもまだ一般化が十分でないことを示しているともいえる。これをさらに DSBG の手法で一般化しようとする、DS マクロオペレータだけでなく普通のマクロオペレータも取り除いたうえで一般化しようということになる。しかし、そうすると述語 DS は、マクロオペレータすべてを用いて探索しなければならず、効率が悪化し DS 概念の操作可能性が低下する。このような、一般化の十分性と操作可能性との間の trade-off<sup>(17)</sup> に対し、どの地点でバランスをとるかは問題であるが、筆者は DS マクロオペレータによる直接解決可能性が妥当なものだと考えている。

また、DS 概念の操作可能性が常に保証されるのかという問題がある。Keller は以下の二つの条件を満たすとき、その概念記述は操作可能であるとしている<sup>(14)</sup>。  
 ①使用可能性 (Usability: 概念記述が問題解決システムにとって使用可能である)、②有効性 (Utility: その概念記述の使用により明記された目的について問題解決システムの効率が向上する)。従来の定義は「使用可能であれば有効である」という暗黙の前提があったのに対し、Keller の定義は使用可能性と有効性を独立したものと考え、さらに有効性を「目的」に依存するものとしている点が非常におもしろい。この Keller の定義に従うと DS 概念は使用可能性は常に満たしているが、有効性は必ずしも保証されていない。なぜなら、DS マクロオペレータが非常に増加した場合、DS マクロによる解決可能性の検証が他のオペレータを用いた場合よりも相対的に効率がよい。つまり有効であ

るとは限らないからである。

最後に、もう一つの重要な問題点として、有害なマクロオペレータが獲得されることがある。有害なマクロは今回の実験では下のマクロオペレータのみであっ

$$R1 * AL \rightarrow 2 * R2 * AL$$

たが、これは 2 次方程式の教示 (完全平方の解き方) の際に抽出される。この有害なマクロオペレータの抽出は完全因果性に問題があるのではなく、EBG の本質的な問題点である領域知識の不完全性によるものである。現在のところ、PiL はこの種のマクロオペレータを人為的に駆除しているが、これもシステムが自動的に検出・修正することが望まれ、今後の大きな課題である。

## 10. ま と め

DS マクロオペレータを用いて、EBG を拡張した一般化手法である直接解決可能性に基づく一般化 (DSBG) を提案し、PiL システムにおける実験によりその有効性を検証した。今後としては、残りの EBG の問題点への対処法、問題状態表現を述語表現に統一することによる PiL システムの汎用化<sup>(18)</sup>などを考えている。

## 謝 辞

方程式教示の参考にした問題集を提供していただいた大阪教育大学付属天王寺高校吉岡大蔵君ならびに有益な議論をしてくれた辻研 AI グループの皆さんに感謝する次第です。

## ◇ 参 考 文 献 ◇

- (1) Mitchell, T. M., Keller, R. M. and Kadar-Cabelli, T.: Explanation-Based Generalization: A Unifying View, *Machine Learning*, 1-1, pp. 47-80 (1986).
- (2) DeJong, G. and Mooney, R.: Explanation-Based Generalization: An Alternative View, *Machine Learning*, 1-2, pp. 145-176 (1986).
- (3) Mitchell, T. M., Mahadevan, S. and Steinberg, L. I.: LEAP: A Learning Apprentice for VLSI, *IJCAI '85* (1985).
- (4) Rosenbloom, P. S. and Laird, E. J.: Mapping Explanation-Based Generalization onto Soar, *AAAI '86* (1986).
- (5) Minton, S. and Carbonell, J. G.: Strategies for Learning Control Rules: An explanation-based Approach, *IJCAI '87* (1987).
- (6) Mostow, J. and Bhatnagar, N.: Failsafe: A Floor Planner that Uses EBG to Learn from its Failure, *IJCAI '87* (1987).
- (7) Hirsh, H.: Explanation-Based Generalization in a Logic-Programming Environment, *IJCAI '87* (1987).
- (8) Kedar-Cabelli, S. T.: Formulating Concept According to Purpose, *AAAI '87* (1987).
- (9) 渡辺, 岩本, 山之内, 松田: VILLA: VLSI 設計知識獲得システム, *情報処理研究会*, AI-86-1 (1986).
- (10) 山田, 安部, 辻: 問題解決における戦略知識学習システム: PiL, *人工知能学会誌*, Vol. 3, No. 2 (1988).
- (11) Mitchell, T. M., Utgoff, P. E. and Banerji, R.: Acquiring and Refining Problem-Solving Heuristic, *Machine Learning*, pp. 163-190—An artificial intelligence approach Vol. 1, Tiago Pub. Co., Palo Alto (1983).
- (12) Mitchell, T. M.: Learning and Problem Solving,



- IJCAI '83, pp. 1139-1151 (1983).
- (13) Silver, B. : Meta-Level Inference, pp. 127-151, North Holland (1986).
- (14) Keller, R. M. : Defining Operationality for Explanation-Based Learning, pp. 482-487, AAAI '87 (1987).
- (15) 山田, 安部, 辻 : 問題解決における学習システム : PiL「人工知能システムの枠組み」シンポジウム (1987).
- (16) 山田, 安部, 辻 : 問題解決における戦略学習システム : PiL—直接解決可能性に基づく一般化—, 情報処理研究会, 88-AI-56 (1988).
- (17) Segre, A. M. : On the Operationality/Generality Trade-Off in Explanation-Based Learning, IJCAI '87 (1987).
- (18) 山田, 辻 : 完全因果性によるマクロオペレータの選択的学習, 情報処理研究会, 88-AI-60 (1988).

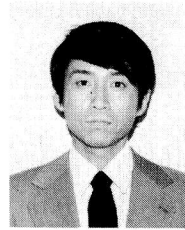
〔担当編集・査読委員 : 林 弘〕

著者紹介



山田 誠二 (正会員)

昭和 59 年大阪大学基礎工学部卒業。昭和 61 年同大学院修士課程修了。現在、同博士課程在学中。人工知能、特に学習、説明に基づく一般化およびマクロオペレータに関する研究に従事。情報処理学会、IEEE 各会員。



安部 憲広 (正会員)

昭和 44 年大阪大学基礎工学部電気工学科卒業。昭和 49 年同大学院博士課程修了。同年、同大学基礎工学部制御工学科勤務。現在、同学部助教授。工学博士。人工知能、ロボティクスなどの研究に従事。情報処理学会、電子情報通信学会、計測自動制御学会各会員。



辻 三郎 (正会員)

昭和 28 年大阪大学工学部卒業。昭和 30 年同大学院修士課程修了。同年、電子技術総合研究所入所。昭和 45 年より大阪大学基礎工学部制御工学科教授。工学博士。現在、人工知能、ロボティクス、コンピュータビジョンなどの研究に従事。情報処理学会、電子情報通信学会、計測自動制御学会、IEEE 各会員。