

初等幾何学の補助線問題におけるフラストレーションに基づく学習

Frustration-Based Learning in Auxiliary-Line Problems in Elementary Geometry

諏訪 正樹^{*1} 元田 浩^{*2}
Masaki Suwa Hiroshi Motoda

*1 東京大学工学部原子力工学科
Dept. of Nuclear Eng., University of Tokyo, Tokyo 113, Japan.

*2 (株)日立製作所基礎研究所
Advanced Research Laboratory, Hitachi, Ltd., Kokubunji Tokyo 185, Japan.

1988年8月19日 受理

Keywords : machine learning, auxiliary-line problem, associative knowledge, knowledge acquisition.

Summary

We have developed a learning system, AUXIL, which has an ability to solve auxiliary-line problems in geometry in an intelligent way.

First, we show that a basic mechanism for producing auxiliary-lines is to associate a certain condition or subgoal in the problem with an appropriate figure-pattern and that AUXIL can produce a right auxiliary-line by making use of associative knowledge, which we call figure-pattern strategies.

Secondly, we proposed a new method, frustration-based learning, which can learn associative knowledge from experiences of solving a variety of auxiliary-line problems. AUXIL simulates the following expert behavior. When an expert tries to solve such a problem, he feels frustration because enough information is not given in a problem space for him to proceed an inference and to find a correct path from given conditions to the goal. Here, he concentrates himself on the conditions or subgoals which have caused frustration. After he has produced an auxiliary-line and made a complete proof-tree, he would learn several pieces of associative knowledge. Each frustration-causing condition or subgoal will constitute the if-part of each knowledge. He will then recognize several lumps of figure-patterns in the proof-tree, each of which has contributed to resolving each frustration. All pieces of geometrical information of each figure-pattern will constitute the then-part of each knowledge.

A figure-pattern strategy has two characteristics as an associative knowledge. One is that its application to problems does not necessarily contribute to successful paths of the problems because it is a mere successful instance in the past experiences. The other is that it can enjoy flexible application to problems under no constraint of their goal-structures because its if-part can be unified, if unifiable, to any partial element of the problems. The second characteristic enables AUXIL to produce an appropriate auxiliary-line by a multiple use of figure-pattern strategies in response to several frustrations occurring in a problem, which is sufficient for making up for the first undesirable characteristic.

1. はじめに

補助線問題とは、新たな線分もしくは点を付け加えなければ解くことのできない平面幾何学の問題を言

う。本論文では、中でも新しい点を一つだけ必要とするタイプの補助線問題を扱うことにする。Fig. 1に典型的な補助線問題を3題示す。これらの問題が補助線を必要としない問題と違う点は、本来必要な点が欠如しているために与条件と証明すべきゴールの間に証明

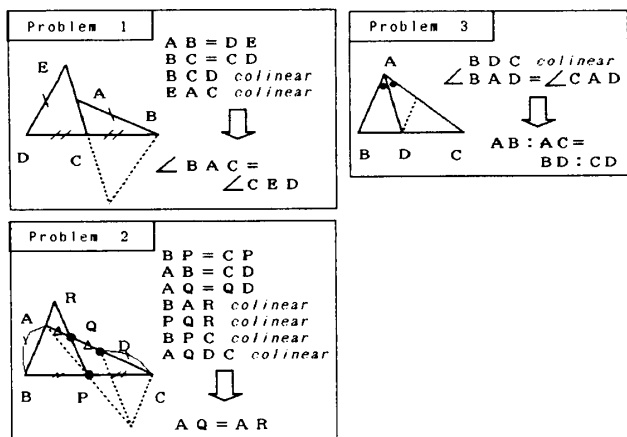


Fig. 1 Typical auxiliary-line problems.

パスの成立を遮るギャップが横たわり、ギャップの両側で定理を前向きにも後向きにも適用できない不連続点が存在することである。妥当な補助線を引く行為は、証明パスの成立を可能にするようにギャップに橋を架けることに相当する。

本論文では“人間が補助線問題に遭遇したとき、過去の経験をどのように利用して正しい補助線を引くのか?”に関して以下のような仮説を立て、それを計算機上に実現する学習アルゴリズムを提唱する。初めて接した補助線問題が Fig. 1 の第 1 問であり、「EC の延長線と B から ED に平行に引いた直線との交点を新しい点として創り出せ」との教示を受けこの問題を証明したとする。その後、Fig. 1 の第 2 問に遭遇したとすると、両者の類似性を感じ取り図の点線のような補助線を生成できるであろう。類似性を感じる理由は、両問題において注目すべき問題要素が同じだからである。そして、第 1 問で体得した(であろう)「補助線生成法」を第 2 問に適用して証明を完成した後に、改めて両者の類似性を強く意識し、適用した「生成法」の記憶が強化されるであろう。このようなプロセスを何度も経験することによって次第にさまざまな補助線問題を解決できるようになると予想される。

本論文の目的は、以下の 2 点を明らかにすることである。

- (1) 補助線を生成するための知識群はいかなるものか。
- (2) (1)の知識をいかにして学習するのか。

2 章、3 章において各々の問題に対する解答を提案する。4 章では、2、3 章で述べる機能を有する学習システム AUXIL によるシミュレーション結果を提示する。5 章では、3 章の学習法と従来の学習法との相違点を示す。

2. 補助線生成の基本メカニズム

1970 年代に定理証明の研究の一環として補助線問

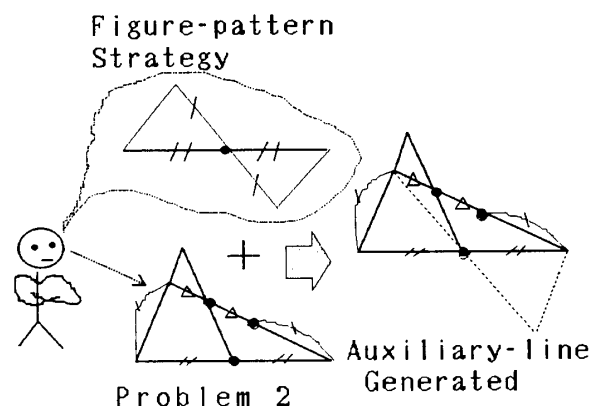


Fig. 2 Basic mechanism for generation of auxiliary-lines.

題を取り扱った試みがある⁽⁵⁾⁽¹⁵⁾⁽¹⁶⁾⁽¹⁸⁾。なかでも、定理証明器による解法を最も具体的に提案したのは R. Wong である⁽¹⁶⁾。彼は、補助線の生成方法は 3 種類に分類できることを示した。しかし、“補助線の生成メカニズムとして異なるものが複数種ある”という立場を取るとすると、その各々に対して異なる学習法を与えなければならず、学習システムとして好ましくない。補助線の生成は一つのメカニズムで説明できるほうが好ましい。

そこで、我々は、問題中の特定の条件もしくはサブゴールに対して、条件反射的に図形パターンを連想することが補助線生成の基本メカニズムであるという仮説を立てた。そのための連想用知識(図形パターン戦略と呼ぶ)は次のように表される。

IF : 特定の条件および(もしくは)サブゴールがあれば

THEN : ある図形パターンを連想せよ。

その IF 部は連想を喚起する問題要素であり、THEN 部は連想の仕方を示す手続きである。連想による補助線生成のプロセスの一例を Fig. 2 に示す。生徒が問題 2 において $BP=PC$ という条件に注目したとしよう。もし

IF : there exists 条件 $(XY=YZ)$ (X, Y, Z は変数)

THEN : 図形パターン (Fig. 2) を連想せよ

という図形パターン戦略を持っているとすれば、問題図形と連想された図形パターンの一部を Fig. 2 のように重ね合わせることによって図形パターンに存在して問題図形にない点および線分が補助線として見えてくるであろう。

このメカニズムは、論理プログラミング言語 Prolog のユニフィケーション技法を用いて機械的に実現することができる。そのためには、幾何学的情報を Prolog の述語で表現する必要がある。幾何学的情報は、図形を構成する構成情報と、構成された対象間に成り立つ

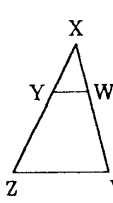
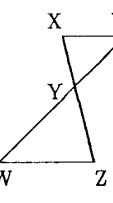
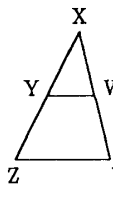
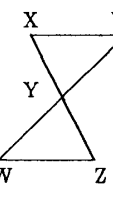
Strategy 1	 <pre> IF ?-subgoal(equal(ratio(seg(X,Y),seg(Y,Z)),P)). THEN ?-make_pattern(s1,[X,Y,Z,W,V]). where make_pattern(s1,[X,Y,Z,W,V]):- (point(X),point(Y),point(Z),point(W), point(V),line([X,Y,Z]),line([X,W,V]), parallel(line(Y,W),line(Z,V)),line([Y,W]), line([Z,V]) → (\$similar(tri(X,Y,W),tri(X,Z,V)), \$equal(ang(X,Y,W),ang(X,Z,V)), \$equal(ang(X,W,Y),ang(X,V,Z)), \$equal(ratio(seg(X,Y),seg(X,Z)),ratio(seg(X,W),seg(X,V))), \$equal(ratio(seg(X,Y),seg(X,Z)),ratio(seg(Y,W),seg(Z,V))), \$equal(ratio(seg(Y,W),seg(Z,V)),ratio(seg(X,W),seg(X,V))), \$equal(ratio(seg(X,Y),seg(Y,Z)),ratio(seg(X,W),seg(W,V)))). </pre>
Strategy 2	 <pre> IF ?-subgoal(equal(ratio(seg(X,Y),seg(Y,Z)),P)). THEN ?-make_pattern(s2,[X,Y,Z,W,V]). where make_pattern(s2,[X,Y,Z,W,V]):- (point(X),point(Y),point(Z),point(W),point(V), line([X,Y,Z]),line([W,Y,V]),line([X,V]), line([W,Z]),parallel(line(X,V),line(W,Z)) → (\$similar(tri(X,Y,V),tri(Z,Y,W)), \$equal(ang(X,Y,V),ang(Z,Y,W)), \$equal(ang(X,V,Y),ang(Z,W,Y)), \$equal(ang(V,X,Y),ang(W,Z,Y)), \$equal(ratio(seg(X,Y),seg(Y,Z)),ratio(seg(V,Y),seg(Y,W))), \$equal(ratio(seg(X,Y),seg(Y,Z)),ratio(seg(X,V),seg(W,Z))), \$equal(ratio(seg(V,Y),seg(Y,W)),ratio(seg(X,V),seg(W,Z)))). </pre>
Strategy 3	 <pre> IF ?-condition(equal(seg(X,Y),seg(Y,Z))). THEN ?-make_pattern(s3,[X,Y,Z,W,V]). where make_pattern(s3,[X,Y,Z,W,V]):- (point(X),point(Y),point(Z),point(W),point(V), line([X,Y,Z]),line([X,W,V]),line([Y,W]), line([Z,V]),equal(seg(X,Y),seg(Y,Z)), equal(seg(X,W),seg(W,V)) → (\$similar(tri(X,Y,W),tri(X,Z,V)), \$equal(ang(X,Y,W),ang(X,Z,V)), \$equal(ang(X,W,Y),ang(X,V,Z)), \$parallel(line(Y,W),line(Z,V)), \$equal(ratio(seg(Y,W),seg(Z,V)),ratio(1,2))). </pre>
Strategy 4	 <pre> IF ?-condition(equal(seg(X,Y),seg(Y,Z))). THEN ?-make_pattern(s4,[X,Y,Z,W,V]). where make_pattern(s4,[X,Y,Z,W,V]):- (point(X),point(Y),point(Z),point(W),point(V), line([X,Y,Z]),line([W,Y,V]),line([X,V]), line([W,Z]),equal(seg(X,Y),seg(Y,Z)), equal(seg(W,Y),seg(Y,V)) → (\$congruent(tri(X,Y,V),tri(Z,Y,W)), \$equal(ang(X,Y,V),ang(Z,Y,W)), \$equal(ang(X,V,Y),ang(Z,W,Y)), \$equal(ang(V,X,Y),ang(W,Z,Y)), \$equal(seg(X,V),seg(Z,W)), \$parallel(line(X,V),line(W,Z))). </pre>

Fig. 3 (a) An example of figure-pattern strategies.

拘束情報の二つに大別できる（以下では、それぞれ、構成情報、拘束情報と呼ぶ）。この区別は3章で述べる学習において重要な役割を持つ。構成情報は、点および線分の存在を表す情報から成り、それぞれ、point述語、line述語で表現する。

一方、拘束情報は線分・角・線分の比が等しいとい

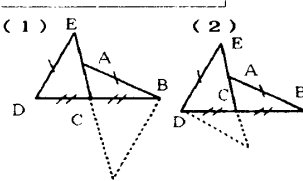
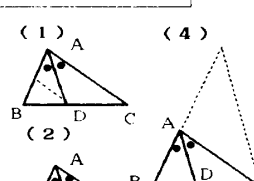
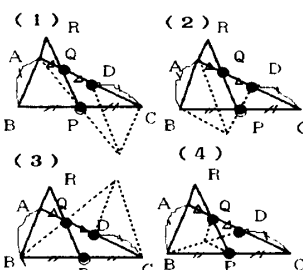
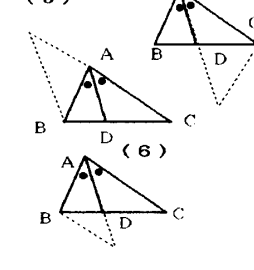
Problem 1 6	Problem 3 6
	
Problem 2 42	
	

Fig. 3 (b) Successful auxiliary-lines produced by figure-pattern strategies.

う情報、二つの線分が平行であるという情報、二つの三角形が合同・相似であるという情報から成り、それぞれ equal 述語, parallel 述語, congruent 述語, similar 述語で表現する。

問題図形は、引き数部の変数が問題中の点を表す数値でインスタンス化された述語の assertion 集合として論理表現する。一方、図形パターン戦略は、

```

IF      : ?-condition (問題条件を表す述語)
or/and : ?-subgoal (サブゴールを表す述語)
THEN   : ?-make_pattern (図形パターン名, 関与する点のリスト)
                
```

where

```

make_pattern (図形パターン名, 関与する点のリスト) :- (図形決定条件) →
(図形に成り立つ条件).
                
```

と表現する。図形決定条件 (if-part) とは、図形パターンの性質を一意に決定する条件である。したがって、make_pattern 述語はある図形パターンを連想するための具体的な手続きを示すマクロオペレータとみなせる。そして、if-part と then-part を合わせた全情報は、その図形パターンが持つすべての性質を表している (全図形パターン情報と呼ぶ)。例えば、Fig. 2 の図形パターン戦略は一例としては Fig. 3 (a) のようになる (Strategy 4)。

AUXIL では問題中の特定条件 (例えば、問題2の BP=PC) と IF 部がマッチする図形パターン戦略を選択し、THEN 部の make_pattern 述語を起動することにより補助線の生成を実現する⁽¹⁷⁾。make_pattern 述語の起動が成功したとき、その if-part の述語で new を含むものが新しい点と補助線の位置を決定す

る情報を表し、then-part の述語で new を含むものが補助線生成後の問題図形に成り立つ新情報を表す。一般に図形パターン戦略の適用を成功させる代入の組合せは複数存在し、その各々が異なる補助線に対応する。

図形パターン戦略は、Greeno が提唱したスキーマ⁽⁶⁾と類似しているが、機械的に補助線生成機構を実現している点、連想用知識として利用される点においてスキーマと異なる⁽¹⁷⁾。

我々の仮説の妥当性を 10 数種の補助線問題において確認した。その一例として、AUXIL に予め Fig. 3 (a) の 4 種類の図形パターン戦略を与えて、Fig. 1 の 3 問を解かせた結果を Fig. 3 (b) に示す。問題名の横に示した数字は、生成され得るすべての補助線パターンの種類を指す。そのうち、証明パスを成立させる有効な補助線パターンのみを図に示した。

ここで、一つ問題点がある。点もしくは与条件を多く含む問題において本メカニズムにより補助線を生成すると、生成される可能な補助線の種類に対する、有効な補助線の種類の比(問題 2 の場合 4/42。これを有効度比と呼ぶ)が極端に小さくなることがある。人間は、図形パターンの使い方に関するヒューリスティクスによりこの問題を解決していることが予想され、4 章でその点に言及する。

最後に、図形パターン戦略の特徴を述べる。戦略の IF 部は問題の部分的な要素にすぎないので、問題中に存在する他の要素と無関係に連想が行える。この性質は、適用性の良さ (Fig. 3 (a) の第 3, 4 の戦略は、問題構造および証明すべきゴールの全く違う問題に適用できる) と、有効度比の低さの両面を図形パターン戦略に与えている。

3. フラストレーションに基づく学習

この章では、補助線問題を解きながら図形パターン戦略を自動的に学習するアルゴリズムを提案する。この学習法は、「戦略の IF 部を構成する問題要素は、問題を解くプロセスの中で注目を集めた要素である」というコンセプトに基づいており、以下の 4 段階のプロセス、①フラストレーション同定プロセス、②補助線生成プロセス、③証明プロセス、④学習プロセスからなる。まず、第一のプロセスで推論における困難点を発見することにより特定の問題要素に注目する。次に、第二、第三のプロセスにおいて完全な証明木を構築する。最後に、第一のプロセスで注目した問題要素を参照しながら証明木の中から図形パターン戦略を抽出する。

3.1 フラストレーション同定プロセス

AUXIL には領域知識として初等幾何学の基本定理のみ(式の変形や比の計算などの代数知識は含まない)を与える。しかし、ギャップがあるため補助線を引かない限り、基本定理を条件もしくはゴールに適用できなくなる箇所(推論停止位置)が現れる。これをフラストレーション要素と呼び、それらを同定するのがこのプロセスの役目である。

AUXIL は前向き・後向き推論の 2 種類の推論方法を持つので、それぞれに対するフラストレーション要素(フラストレーション条件とフラストレーションサブゴール)を以下に定義する。

〔定義 1〕 (フラストレーション条件): 与条件および前向き推論により得られた問題条件のうち、初等幾何学の基本定理を前向きに適用できない拘束条件(構成条件はフラストレーション条件にならない)をフラストレーション条件 (F 条件) という。(定義終了)

F 条件は、条件側から伸びる証明木の最先端ノードを形成する。このフラストレーションは、「どのような条件を使えばよいのかわからずに悩む」ことに対応する。

フラストレーションサブゴールを定義するには準備が必要である。

〔定義 2〕 (確定サブゴール・不確定サブゴール): あるゴール(サブゴール)に基本定理を後向きに適用して得られるサブゴールは 2 種類に大別できる。引き数に変数を含まないものを確定サブゴール、少なくとも一つ変数を含むものを不確定サブゴールという。(定義終了)

不確定サブゴールが発生するのは、ヘッド部に現れない変数をボディ部に持つような基本定理を後向きに適用したときである。

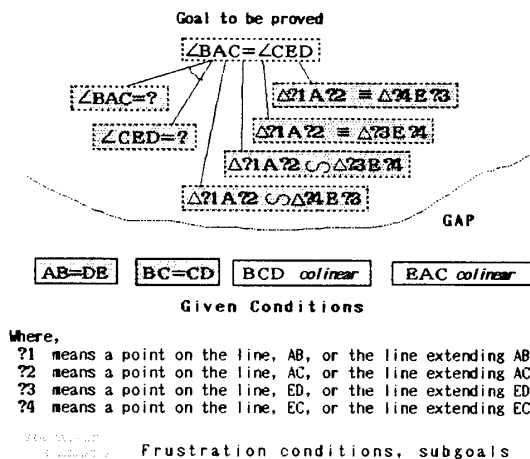
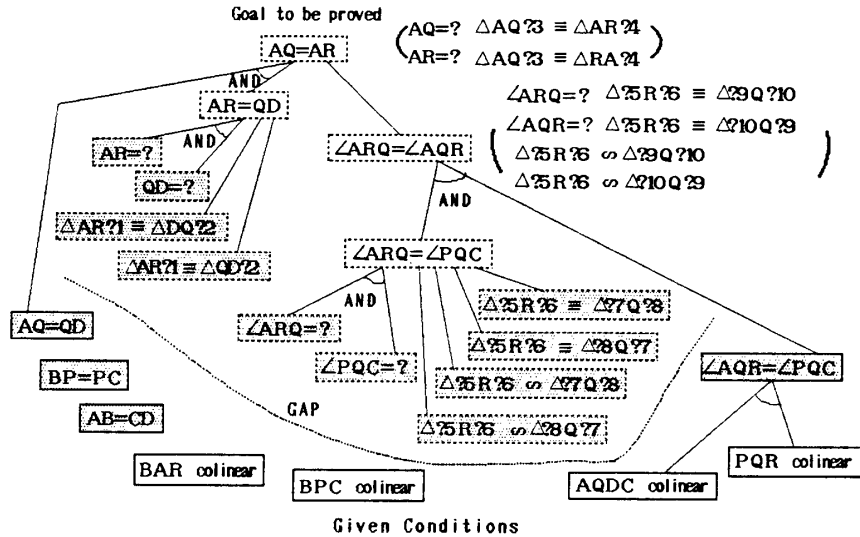


Fig. 4 A frustration tree (in case of Problem 1 in Fig. 1).



Where,
 ?5 means a point on the line, RA, or the line extending RA through A,
 ?6 means a point on the line, RQ, or the line extending RQ through Q,
 ?7 means a point on the line, QP, or the line extending QP through P,
 ?8 means a point on the line, QC, or the line extending QC through C,
 ?9 means a point on the line, QA, or the line extending QA through A,
 ?10 means a point on the line, QR, or the line extending QR through R.

Frustration conditions, subgoals

Fig. 5 A frustration tree (in case of Problem2 in Fig. 1).

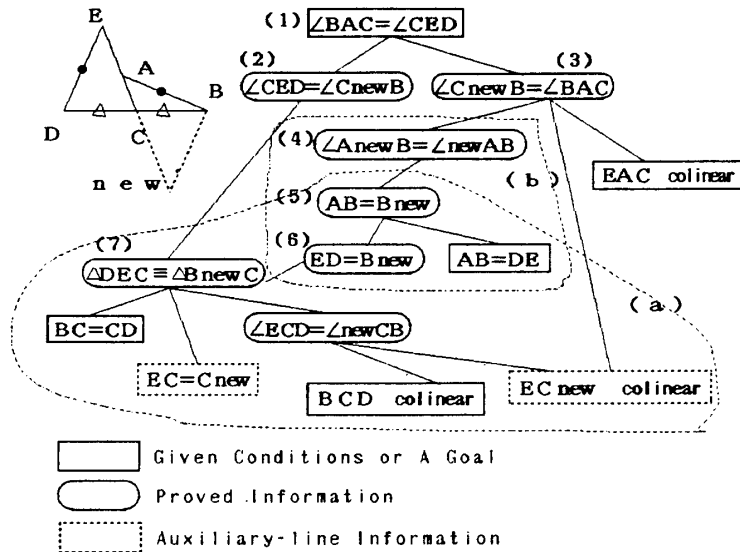


Fig. 6 A proof tree (in case of Problem1 in Fig. 1).

〔定義3〕 (一時的確定サブゴール):あるサブゴールの下に一組のand-並列のサブゴールが発生し、それらがすべて不確定サブゴールである状況を想定しよう。不確定サブゴールには、与条件もしくは問題条件にユニファイ可能なものと不可能なものがある。いま発生したサブゴールのうち幾つかがユニファイされた結果、残りのサブゴールが確定サブゴールになる可能性がある。これを一時的確定サブゴールという。(定義終り)

〔定義4〕 (同レベルサブゴール):一般に一つのサブゴールには複数の定理が適用可能で、したがって

複数組のサブゴールが発生する。それらを互いに同レベルサブゴールという。(定義終り)

ここで後向き推論に関して次の仮説を立てる。

《仮説1》 後向き推論は、確定サブゴールから下のみ連鎖的に続行されるものとする。不確定サブゴールに適用しない理由は、新たに生まれる下位サブゴールはより多くの不確実性をもつ可能性があり、実質的にサブゴールとして機能しないことが予想されるからである。(仮説終り)

この仮説により不確定サブゴールは、ゴール側から伸びる証明木の最先端ノードを形成することになる。

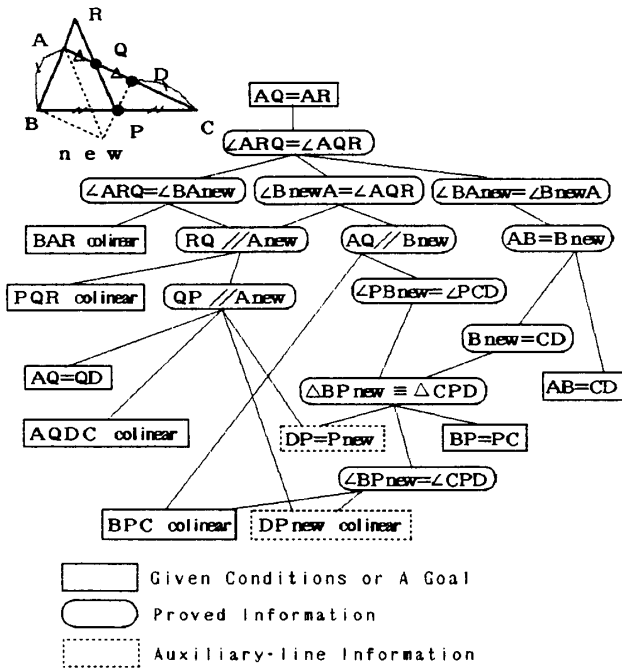


Fig. 7 A proof tree (in case of Problem2 in Fig. 1).

〔定義5〕 (フラストレーションサブゴール): 不確定サブゴールのうち、以下の二つの要件を満たすものをフラストレーションサブゴール (Fサブゴール) と定義する。①同レベルサブゴールに、確定サブゴールも一時的確定サブゴールも存在しないもの、②それ自身が一時的確定サブゴールにはならないもの。(定義終り)

同レベルに確定サブゴールを持つような不確定サブゴールは、仮説1により確定サブゴールの下へ推論が展開されるので、Fサブゴールとは考えない。ただし、AUXILは、それらの不確定サブゴールと、ある問題条件とユニファイして一時的確定サブゴールになる前の不確定サブゴールを、Fサブゴール候補として記憶する機能を有する。

AUXILがFig. 1の問題および2を解く過程で発生させたF要素をFig. 4, Fig. 5に示す(この部分的証明木をフラストレーション木と呼ぶ)。影付きのノードはF要素を表す。また、ノードの横に書かれた不確定サブゴールはフラストレーション候補である。

3・2 補助線の生成

AUXILは、図形パターン戦略を利用して補助線を生成するか、または、適用できる図形パターン戦略

* AUXILは後者の場合に教示を受けなくても自ら補助線を生成する機能を有している⁽¹⁷⁾。しかし、補助線生成プロセスは、完全な証明木を構築するための1ステップにすぎないので、どちらの方法で補助線を生成しても学習プロセスにおける図形パターン戦略の抽出には影響を及ぼさない。本論文では教示によって補助線を生成するものとする。

を持たない場合はユーザーにその引き方の教示を受ける。後者の場合、補助線の位置を表す述語の assertion をユーザーから与えてもらう。

3・3 証明プロセス

基本定理を駆使して再び完全な証明木の形成を試みる。その過程で新たにF条件が発生した場合、補助線生成プロセスに戻って補助線を引き直す(補助線がユーザーの教示により生成された場合にはこのような事態は起こらない)。問題1, 2の正しい証明木の一例をFig. 6, Fig. 7に示す。

3・4 学習プロセス

このプロセスにおいて、AUXILは証明木の中から複数個の図形パターン戦略を抽出する。以下の四つのサブプロセス、①フラストレーション木分析サブプロセス、②フラストレーション解消サブプロセス、③図形内限定推論サブプロセス、④図形パターン一般化サブプロセスを経て、木のどの部分を抽出すべきかが決定される。

〔1〕 フラストレーション木分析サブプロセス

このプロセスでは、F木と証明木を比較しF木の中で証明に貢献した部分を切り出すと共に、解消されたF要素を同定する。そのためのタスクの詳細は文献(17)に詳しい。これにより得られる木を精製F木と呼ぶ。問題2の精製F木をFig. 8に示す(角ARQ=角AQRから下のサブゴール群が削除され、代わりにそのFサブゴール候補が新たなFサブゴールになる点に注意)。問題1のF木(Fig. 4)はこのプロセスによる変化を受けない。

〔2〕 フラストレーション解消サブプロセス

前サブプロセスで同定されたF要素が、補助線の生成によっていかに解消されたかを調べ、証明木から図形パターンを抽出するプロセスである。そのために、AUXILは、精製F木の中の各F要素ごとに以下の二つのタスクを行う。

(1) F要素を解消するのに直接貢献した基本定理を、問題中の定数でインスタンスシートされたまま抽出する。F条件を解消するのに貢献した定理とは、その条件をボディ部に含む定理であり、Fサブゴールを解消するのに貢献した定理とは、そのサブゴールをヘッド部とする定理である。

(2) (1)で抽出された定理に含まれている構成情報を抽出する(それらの構成情報で形成される図形パターンを取り出すことに相当する)。

一例として、問題2のF条件BP=PCに対して抽

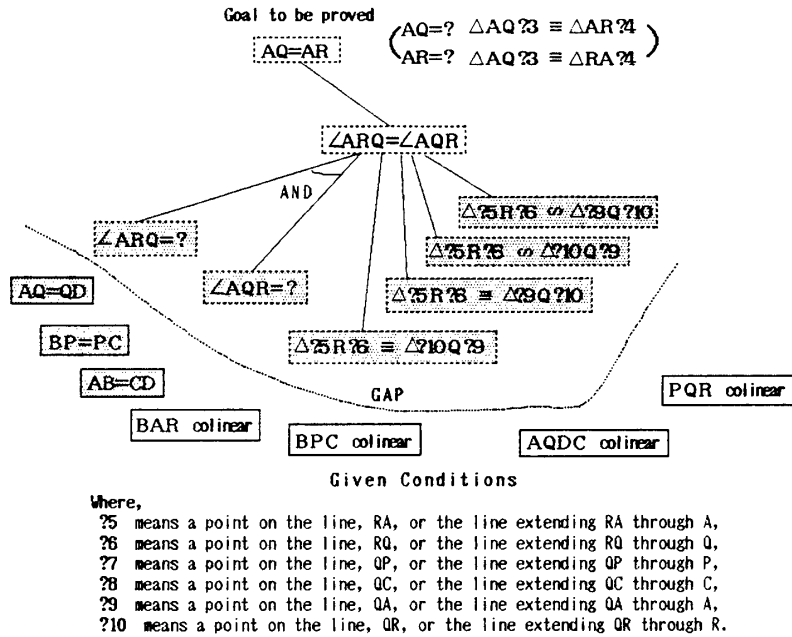


Fig. 8 A revised frustration tree (in case of Problem 2 in Fig. 1).

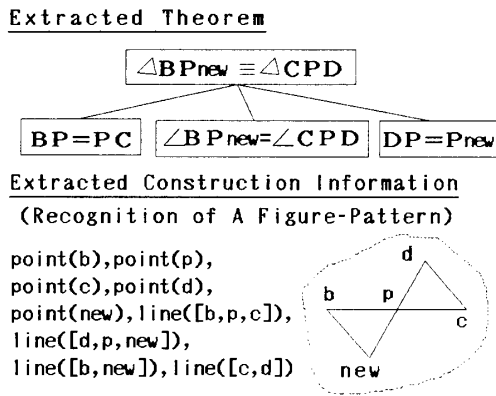


Fig. 9 Resolution subprocess.
 (in learning, motivated by a frustration condition, $BP=PC$, of Problem 2)

出された定理およびそれに含まれる構成情報を Fig. 9 に示す. なお, 解消されない F 要素に関しては何も行わない.

上記の二つのタスクを各 F 要素に関して行うことにより, AUXIL はフラストレーションの解消の仕方だけでなくそれが解消された状況をも憶える.

〔3〕 図形内限定推論プロセス

このサブプロセスで, AUXIL は, 3・4 節〔2〕で抽出した図形が持つすべての情報を前向き推論により収集する. 解消プロセスで抽出した情報のみを使って(その図形の中だけで)推論を行うため, 「限定」推論と呼ぶ. Fig. 9 の図形で限定推論を行うと, $B_{new}=CD$, 角 $B_{new}P =$ 角 CDP , 角 $newBP =$ 角 DCP , $B_{new} // CD$ という情報が得られる (Fig. 10 (a)). 三角形 $BPD \equiv$ 三角形 CP_{new} , $BD=C_{new}$ などは, BD 一直線, C_{new}

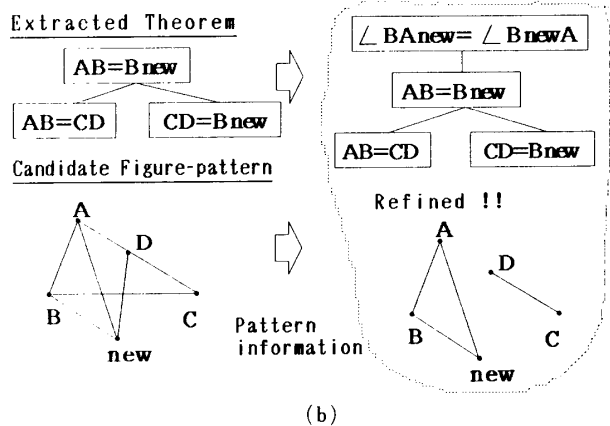
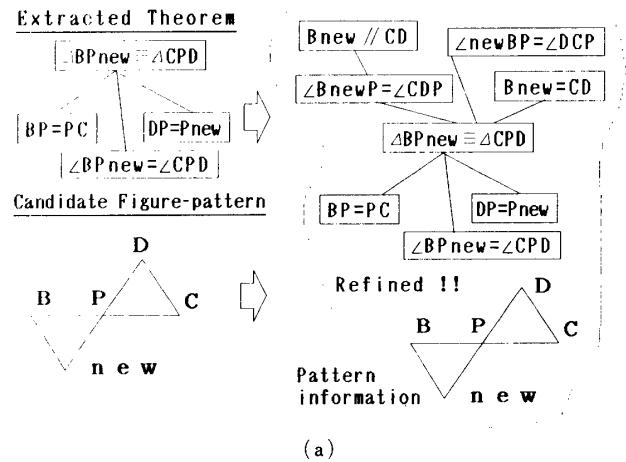


Fig. 10 Limited forward-reasoning subprocess.
 (in learning, motivated by a frustration condition,
 (a) $BP=PC$, of Problem 2, (b) $AB=CD$, of Problem 2).

一直線という構成情報が存在しないため得られない.
 図形内限定推論を証明プロセスとは別に設けること

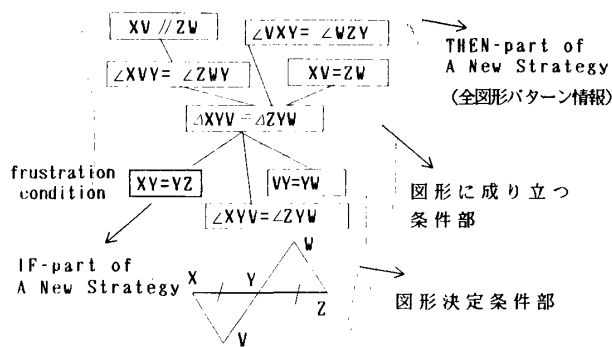


Fig. 11 Figure-pattern generalization subprocess. (in learning, motivated by a frustration condition, BP=PC, of Problem 2).

は冗長であるとも見えるかもしれないが、このプロセスは三つの重要な役割を持つ。

- (1) 当問題の証明木の成立には貢献しなかった情報を含めて、抽出された図形パターンが持つあらゆる情報を調査すること。なぜなら、貢献しなかった情報(例えば角 newBP=角 DCP)は、将来この図形パターン戦略が連想されるときには有用な情報であるかもしれないからである。「AUXILは正しい証明のシーケンスを記憶するのではなく、連想の対象となる図形パターンを記憶する」という、我々の学習法の目的に照らし合わせて考えれば、この役割は自然である。
- (2) この前向き推論において「言及」されない構成情報を削除することにより、抽出した図形パターンを洗練すること。「言及される」とは、それ自身がこの推論による証明木のノードになっているか、または証明木中の少なくとも一つの拘束情報を構成する情報であるかの状況を指す。Fig. 10(a)の例では、図形パターンは洗練を受けていないが、Fig. 10(b)の例(問題2の AB=CD というF条件に対する図形内限定推論)では、かなり多くの構成情報が取り除かれている。
- (3) 得られた図形パターンを一般化するための基準となること。これに関しては3・4節[4]で説明する。

AUXILは、解消プロセスと図形内限定推論プロセスによって、各F要素に対して抽出された図形パターンの全図形パターン情報を得る。

[4] 図形パターン一般化プロセス

全図形パターン情報を図形パターン戦略として将来の問題にも適用できるように一般化する手法は、説明に基づく学習⁽¹²⁾において説明木を一般化する方法と同じで、限定推論プロセスの証明に使用した定理中の

Strategy A	
Same as Strategy 4 in Fig.3(a)	
	<pre> IF ?-condition(equal(seg(X,Y),seg(Z,W))) THEN ?-make_pattern(p.b,[X,Y,Z,W,V]). where make_pattern(p.b,[X,Y,Z,W,V]):- (point(X),point(Y),point(Z),point(W), point(V),equal(seg(X,Y),seg(Z,W)), equal(seg(X,Y),seg(W,V)),line([X,Y]), line([Z,W]),line([W,V]),line([Z,V])) -> (\$equal(seg(Z,W),seg(V,W)), \$equal(ang(W,Z,V),ang(W,V,Z))). </pre>
	<pre> IF ?subgoal(equal(ang(V,X,Y),P)). THEN ?-make_pattern(p.a,[X,Y,Z,W,V]). where make_pattern(p.a,[X,Y,Z,W,V]):- (point(X),point(Y),point(Z),point(W), point(V),congruent(tri(X,Y,V),tri(Z,Y,W)), line([X,Z]),line([W,Y,V]), line([X,V]),line([W,Z])) -> (\$equal(seg(X,Y),seg(Y,Z)), \$equal(seg(W,Y),seg(Y,V)), \$equal(seg(X,V),seg(Z,W)), \$parallel(line(X,V),line(W,Z)), \$equal(ang(X,Y,V),ang(Z,Y,W)), \$equal(ang(X,V,Y),ang(Z,W,Y)), \$equal(ang(V,X,Y),ang(W,Z,Y))). </pre>
	<pre> IF ?-subgoal(equal(ang(X,Y,Z),P)). THEN ?-make_pattern(p.d,[X,Y,Z]). where make_pattern(p.d,[X,Y,Z]):- (point(X),point(Y),point(Z), equal(seg(X,Y),seg(X,Z)), line([X,Y]),line([X,Z]),line([Y,Z])) -> equal(ang(X,Y,Z),ang(X,Z,Y)). </pre>

Fig. 12 Acquired figure-pattern strategies from Problem 1.

変数の束縛関係を保存したまま証明木の各ノード(幾何学的情報)を変数化するという方法を採用する。Fig. 10(a)の全図形パターン情報を一般化して得られる木をFig. 11に示す。

[5] 図形パターン戦略の形成

図形パターン戦略の形成の仕方をFig. 11に示す。その図形パターンを抽出する動機になったF要素を戦略のIF部とする。そしてTHEN部(make_pattern述語の定義)は全図形パターン情報により構成する。make_pattern節のボディ部のif-partは、図形パターンを形作る構成情報と解消プロセスで抽出した基本定理のボディ部の情報から構成する。また、then-partは全図形パターン情報の残りの情報から構成する。問題解決の経験の中で注目を集めたF要素が学習の契機と基準を与えているので、この学習法をフラストレーションに基づく学習法と呼ぶ。この学習法は、人間が悩み抜いた事柄ほどその解決法をそのときの状況と一緒によく記憶するという傾向を反映するものである。

AUXILでは、各戦略に有効性を表す尺度として使用頻度値という属性を与えている。さまざまな補助線問題から全く同じ戦略が学習されることが多々あり、その場合に戦略の使用頻度値を1ずつ増していく。

Strategy A	
Same as Strategy A in Fig.3(a) but different in <i>make_pattern</i> procedure	
Strategy B	
Same as Strategy B in Fig.8	
Strategy E	
Same as Strategy 3 in Fig.3(a)	
Strategy F	
	<pre>IF ?-subgoal(equal(ang(X,Y,Z),P)). THEN ?-make_pattern(p_f,[X,Y,Z,W,V]). where make_pattern(p_f,[X,Y,Z,W,V]):- (point(X),point(Y),point(Z),point(W), point(V),parallel(line(X,V),line(Y,Z)), line([W,X,V]),line([X,V]),line([Y,Z])) → \$equal(ang(X,Y,Z),ang(W,X,V)).</pre>
Strategy G	
	<pre>IF ?-subgoal(equal(ang(X,Y,Z),P)). THEN ?-make_pattern(p_g,[X,Y,Z,W,V,U]). where make_pattern(p_g,[X,Y,Z,W,V,U]):- (point(X),point(Y),point(Z),point(W), point(V),point(U), parallel(line(X,Y),line(W,V)), parallel(line(Z,Y),line(U,V)),line([X,Y]), line([Y,Z]),line([W,V]),line([U,V])) → \$equal(ang(X,Y,Z),ang(W,V,U)).</pre>
Strategy H	
	<pre>IF ?-condition(equal(seg(X,Y),seg(Z,W))). THEN ?-make_pattern(p_h,[X,Y,Z,W,V,U,S]). where make_pattern(p_h,[X,Y,Z,W,V,U,S]):- (point(X),point(Y),point(Z),point(W), point(V),point(U),point(S), equal(seg(X,Y),seg(Z,W)), equal(ratio(seg(X,Y),seg(V,U)),ratio(1,K)), equal(ratio(seg(Z,W),seg(V,S)),ratio(1,K)), line([X,Y]),line([Z,W]),line([U,V]),line([V,S]),line([U,S])) → (\$equal(seg(V,U),seg(V,S)), \$equal(ang(V,U,S),ang(V,S,U))).</pre>

Fig. 13 Acquired figure-pattern strategies from Problem 2.

AUXIL が学習した図形パターン戦略の例を挙げる。問題 1 と 2 を Fig. 6 および 7 のように解いた場合、AUXIL はそれぞれ Fig. 12 の戦略 A, B, C, D および Fig. 13 の戦略 A, B, E, F, G を学習する。問題 2 を Fig. 3 (b) の (4) の補助線により解いた場合には、Fig. 13 の戦略 E, F, H を学習する。

戦略 B は F 条件 $AB=CD$ (問題 2) に対して学習されたものである。図形内限定推論プロセスにおける図形の洗練によって、戦略 B は三角形 ZWV に対する辺 XY の相対的位置を拘束する情報 (問題 2 では A, D, C は一直線上にあった) を奪われている。したがって、戦略 B を補助線生成用に適用しても (2 章の方法)、*make_pattern* 述語の図形決定条件部の情報の拘束性が弱いため、新しい点の位置を一意に決定できない。このような戦略を消極的戦略と呼ぶことにする。それに対し、補助線の位置を一意に決定できるだけの拘束性を持つ戦略を積極的戦略と呼ぶ。消極的戦略の役割については次章で詳説する。戦略 A, C, E は積極的戦略であり、残りは消極的戦略である。

戦略 H は戦略 B に酷似している。人間なら両者を統

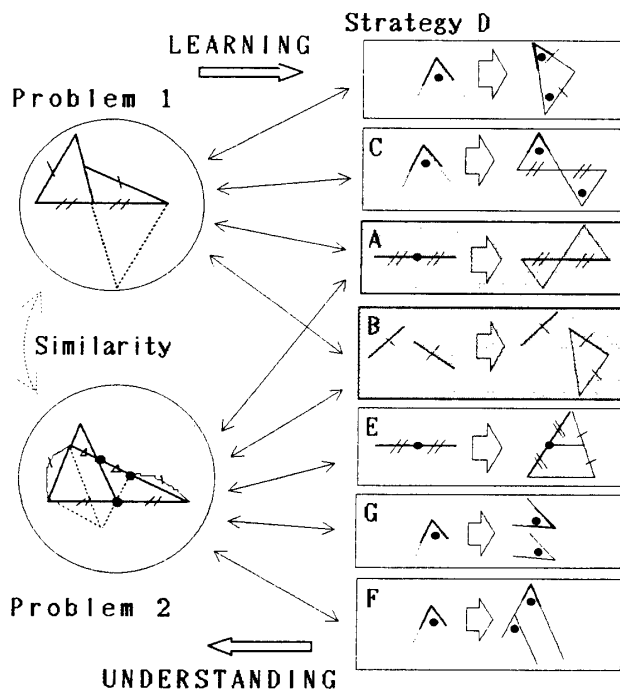


Fig. 14 A meaning of frustration-based learning.

合して、より一般的な戦略「もし $XY=WZ$ という F 条件が存在すれば、XY と WZ を同じように拡大または縮小して得られる辺 VU と VS からなる二等辺三角形を連想せよ」を得ることが予想される。このような高階の抽象的なコンセプトの学習については Shavlik⁽¹⁴⁾ らの研究があるが、AUXIL はその機能は持たない。

最後に、フラストレーションに基づく学習によって問題を経験することの意味合いについて論ずる。Fig. 14 に見られるように、AUXIL は問題 1 から四つの戦略、問題 2 から五つの戦略を学習する。これは、「AUXIL はフラストレーションに基づく学習を行う行為を通して、問題 1 を四つの図形 (各戦略の THEN 部の図形) の重ね合わせとして認識し、問題 2 を五つの図形の重ね合わせとして認識している」ことを意味する。これらの戦略のうち、二つの戦略が両問題に共通していることに注意しよう。共通する戦略 (したがって共通する図形) があるからこそ生徒は両問題に類似性を感じるであろう。これは学習に際して図形を意識することの正当さを語るものである。したがって、フラストレーションに基づく学習は、連想用知識である図形パターン戦略を学習する方法であるのみならず、問題を大局的に把握するという人間の能力の一面をとらえた「理解の枠組」ともみなせる。

4. 学習システム AUXIL

この章では、図形パターン戦略を獲得した AUXIL が、その学習を活かして新たな問題において知的に補

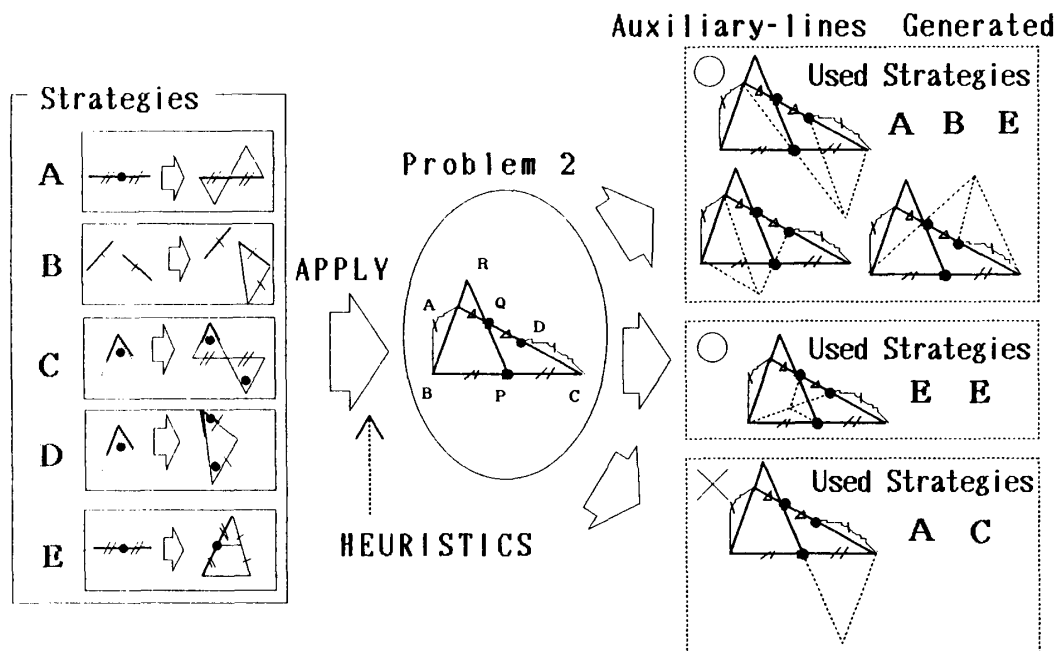


Fig. 15 Multiple use of strategies.

助線を生成する例を示す。「問題1を経験済みで戦略A, B, C, Dを所有しており、かつ他の問題から戦略Eを学習済みである状態で問題2を解く」という状況を想定する。

AUXILは、まず問題分析プロセスでFig.5に示すF木を生成する。次に戦略を適用して補助線を生成する。複数の戦略の適用に関して以下のヒューリスティクスを使用する。

[ヒューリスティクス] できるだけ多くのフラストレーションが解消されるような補助線を与える複数の図形パターン戦略を併用せよ。

このヒューリスティクスは、以下の手続きに従って実現できる。

- (1) まず、ある戦略をあるF要素に対して適用して仮の補助線を生成する(2章の方法)。
- (2) (1)で得た新しい問題図形の中に他の図形パターンを見いだすことができ、かつ他の一つのF要素が解消されるかどうかを調べる。この使い方を消極的適用と呼ぶ。もし、タスク2が成功すれば、他のF要素について手続き(2)を繰り返す。手続き(1)以外の図形パターンが見つからなかったら、(1)に戻って仮の補助線を引き直す。
- (3) 手続き(1), (2)を終えた後に二つ以上のF要素が解消されていたら、その仮の補助線を有望な補助線として採用する。

3章で示唆した消極的戦略の使い方は、消極的適用法に限られる。消極的戦略も含めて複数の戦略を併用することは、それによって有望な少数の補助線だけを選び出すという人間の効率的な問題解決法をシミュ

レートできるという点で、非常に重要な機能である。Fig.15に戦略の併用による問題解決の例を示す。F条件 $BP=PC$ に戦略Aを、 $AQ=QD$ に戦略E、 $AB=CD$ に戦略Bを適用できるような補助線は、Fig.3(b)の(1), (2)の二つの補助線だけに限られる。また、F条件 $AQ=QD$ に戦略A、 $BP=PC$ に戦略E、 $AB=CD$ に戦略Bを適用できるような補助線は、Fig.3(b)の(3)の補助線だけである。 $BP=PC$, $AQ=QD$ の両F条件に戦略Eを適用できるような補助線はFig.3(b)の(4)だけである。ヒューリスティクス1によって偶然有効でない補助線が得られることもある。戦略AをF条件 $BP=PC$ に、戦略CをFサブゴール角 $AQR=?$ に適用して得られる補助線(RPの延長線とCを通過してRBに平行な直線の交点をnewとする)がその例であるが、これは恐らく人間も犯す誤りであろう。

2章で言及した問題2における有効度比の低さは、戦略A, Eを単独で利用していたからであることは明白である。そして、それを克服する手段として複数の戦略を併用することは絶大なる効果を発揮する。併用が可能なのは、IF部が問題中の部分的要素とユニファイ可能で容易に連想が行えるという図形パターン戦略の特長のおかげである。

最後に、本学習法の効果で知的な処理が可能になるのは、補助線の生成の仕方に関する部分だけであることを断っておく。幾何学の問題を計算機上で証明させる際に避けられない困難点(例えば、角度の等価性の判定に関してAUXILではその判断手続きを定義したequivalent述語を用意している)をより人間らしい方法で克服することは、今後の研究に期待する。

5. 従来の学習法との比較

ここで、フラストレーションに基づく学習 (FBL) と、Explanation-based learning (EBL) ⁽⁴⁾⁽¹⁰⁾⁽¹²⁾⁽¹³⁾, マクロオペレータ ⁽³⁾⁽¹¹⁾, チャンキングメカニズム ⁽⁸⁾ を比較する。後者の三つの学習法は、一般化の方法、学習の対象になるターゲットの選び方にこそ差が見られるが、いずれも証明木中のターゲットから下のすべてのサブゴール構造を一般化、あるいはチャンキングすることによってあるコンセプト (target concept) を獲得するという学習法である。これらを広い意味での EBL システムと呼ぶ。EBL システムによって学習される知識 (EBL 知識と呼ぶ) は、IF-THEN 形式で表すことができ、IF 部の要素は THEN 部の target concept を証明するための十分条件である。

これに対して、フラストレーションに基づく学習法は、連想の仕方に関する知識を獲得するためのものである。したがって、FBL 知識は、

IF X THEN Y

where $Y :- A1 \rightarrow B1, Y :- A2 \rightarrow B2, \dots, Y :- An \rightarrow Bn.$

という形態をとる。IF 部は連想を喚起する要素 (いつその知識を使えばよいかを示す要素) であり、THEN 部は連想の手続きである。FBL 知識は、「X という要素で行き詰まったときには、 A_i を発見し (もしくは創り出し) なさい、そうすれば、 B_i という新しい情報が生まれる」という意味を持つ。

FBL が EBL と異なる点は、FBL 知識がそれを用いつ使うべきかに関する情報を持っていること、そして FBL システムが学習するのは、証明のシーケンスではなく連想される対象物であることである。したがって、FBL システムは連想を必要とする問題を連想された対象物の重ね合わせとして理解するという能力を持つ。初等幾何学の補助線問題はその種の問題の典型例であると考えられる。

また、FBL 知識には EBL 知識にない生来の欠点がある。それは、FBL 知識は過去の問題の経験の中でフラストレーションの解消に成功した単なる成功事例にすぎず、IF 部と THEN 部のつながりは EBL 知識の場合と違って検証されていないため、将来の問題に適用されたときにその解決に貢献する保証がないという点である。しかし、連想用知識は本来検証できない、あるいはされるべきものではなく、単に過去に頻繁に生じたことによって強化される類の知識であろう。その意味で、FBL のコンセプトは case-based learn-

ing ⁽²⁾⁽⁷⁾ のそれに似ている。そして、FBL は過去の事例データベースの作り方の一例を提供しているとも言える。AUXIL は、上記の連想用知識独特の欠点をカバーするために、使用頻度値の付与および図形内限定推論プロセスにおける全図形パターン情報の収集という機能を持つ。

EBL 知識は、補助線問題のように連想を必要とする問題を人間と同じように知的に解くには適していないことを説明する。EBL システムが問題 1 の経験から学習し、その後で問題 2 を解こうとするという状況を想定しよう。問題 1 のゴールのみを学習のターゲットとして選ぶ EBL システムと、SOAR ⁽⁸⁾ のようにすべてのサブゴールをターゲットに選ぶ EBL システムを考える。まず、前者が学習する EBL 知識は、問題 1 の全サブゴール構造を一般化したものになっており (Fig. 6 の中の 'EAC colinear' というノードの拘束を受ける)、その全構造を含まない問題 2 に適用することはできない。次に、問題 1 の部分構造をも学習できる後者の EBL システムについて考える。それは Fig. 6 の (1)~(7) の各サブゴールをターゲットにして学習する。例えば、(5) の target-concept は、

IF : $XY=YZ, XYZ$ 1 直線, $WY=YV, WYV$ 1 直線, $XW=ST$

THEN : $ZV=ST$ (X, Y, Z, W, V, S, T は変数)

である。EBL 知識は、それをどのような状況で使うべきかに関する情報を持っていないので、「(1)~(7) のどれを問題 2 で使うべきか」、「前向き・後向きのどちらの方法で使うべきか」、「どのようなユニフィケーションを行うべきか」などの問題が発生する。例えば、(4) または (5) の EBL 知識を $\{X/b, Y/p, Z/c, W/a, V/new, S/c, T/d\}$ という代入によって問題 2 の条件に前向きに適用すれば、Fig. 3 (b) の (1) の補助線が得られるのであるが、すべての可能性の中からの適用の仕方を探し出すことは非常に効率が悪く、しかもこの方法で解答が得られたとしても連想により補助線を生成したことにならない。この不都合は、EBL 知識が FBL 知識の IF 部 (連想を喚起する条件) に相当する情報を持たないこと、すべてのサブゴールをターゲットにするため、あまり重要でない知識をも生成すること、に起因する。

しかし、問題 1, 2 は全ゴール構造は違っていても同じような連想を必要とする問題であるので、問題 1 を経験したことの無い生徒よりも経験済みの生徒のほうが問題 2 の補助線を発見しやすいと予想される。以上の点で、EBL システムは補助線問題のような問題には不向きである。

証明構造の異なる問題間で同じ戦略を共用できるという性格は、図形パターン戦略の適用性の良さと無関係ではない。「target となるサブゴールの下位の全証明構造を憶える」という EBL 原則と、「フラストレーションの解消に関与した部分だけ憶える」という FBL の原則の違いが学習される知識の適用性に違いを生む。問題 1 を例に説明する。(5) の EBL 知識は、Fig. 6 の点線部分 (a) に含まれる情報に拘束されたマクロオペレータであり、 $ED=B_{new}$ という問題要素が三角形の合同によって証明されなければならないという拘束性を持つ。これに対し、AUXIL が $AB=DE$ という F 条件から学習する FBL 知識は、Fig. 6 の点線部分 (b) であり (Fig. 12 の戦略 B)、「学習者は、 $AB=DE$ という条件の使い方 (DE に等しくかつ AB と 1 点を共有するような辺を得て、二等辺三角形を形成する)に興味があるのであって、 $ED=B_{new}$ がいかにして得られるかに深い関心はない」という考え方を根底に持つ。戦略 B は知識 (5) よりも条件の拘束性が少なく、よって適用性に優れている。FBL は証明木の中のどの部分を一つの戦略として抽出すべきかを決定する妥当な尺度を有していると結論できる。

最後に、学習の尺度としてのフラストレーションの導入に関して議論する。FBL では、フラストレーションを定義することによって、問題解決中に注目を集める箇所を同定する。注目要素の同定が学習に直接重要な影響をもたらすという学習研究は数少ない。最も代表的なものは Lebowitz の UNIMEM⁽⁹⁾ であろう。UNIMEM では、説明プロセス (explanation process) を学習者の興味に従って制御できるように interest heuristics を導入しており、これは一種の注目箇所同定用の指標である。しかし、UNIMEM では、interest heuristics を学習者の興味を引かない子細部分を枝刈りするために利用しているのに対し、AUXIL では、フラストレーションを学習の契機および知識の大きさを決定する尺度として利用しており、両者は異なる。

6. おわりに

補助線問題において正しい補助線を生成するためには、証明パス中のギャップを埋める必要があり、問題中の特定要素に喚起される適切な連想を行って初めて可能になる。この連想行為は、それが問題を解くために貢献するかどうかにして深く考慮せずに行う一種の楽観的前向き推論であるとみなせる (Anderson⁽¹⁾ は、この種のデータ駆動推論が専門家と素人の差異を生む一つの要因であることを示唆している)。

本論文では、連想用知識を学習し困難点を克服する方法 (フラストレーションに基づく学習法) を提唱した。連想用知識は、

IF (特定問題要素) THEN (連想行為)
という形式を持つ知識である。以下に、本学習法の特徴を要約する。

- (1) 特定問題要素は、F 同定プロセスで発生した F 要素であり、これが学習の契機になり、更にこの連想を喚起する条件になる。フラストレーションの導入は、連想用知識を学習するのに必要不可欠である。
- (2) 連想行為部は、図形パターンを問題中に創り出す (あるいは発見する) ためのマクロオペレータである。
- (3) 幾何学的情報は、図形を構成する低次元の情報 (学習者の目に訴える情報) と構成された対象間の拘束を示す高次元の情報に分類され、この分類は解消プロセスにおいて重要な役割を果たす。
- (4) 解消プロセスにおいては、解消に貢献した基本定理だけでなく関与する構成情報のすべての候補をも取り込んで図形パターンの候補を形成する。この原則が証明木の中のどの部分を抽出すべきかを決定する尺度になる。
- (5) 図形内限定推論プロセスの役割は、連想される対象物の全図形パターン情報を収集すること、無関係な構成情報を取り除いて図形パターンを洗練することであり、これは証明プロセスとは独立したものである。
- (6) 図形パターンの一般化は、図形内限定推論プロセスで使用した基本定理に基づいて行われる。この手法は EBL の一般化法を踏襲したものである。
- (7) フラストレーションに基づく学習を通して問題を解くことは、学習によって得た戦略の THEN 部にあたる図形パターン (連想対象物) の重ね合わせとして問題を理解することを意味する。

補助線問題は、「連想により適切な問題要素を付け加えて初めて解決可能になる問題」と一般化でき、初等幾何学特有の問題ではない。人間が日常生活の中で行うあらゆる「理解」行為は、常識的に正しいと思われる連想行為に支えられている。例えば、人間どうしの会話が内容の省略にもかかわらず成立しているのは、相手の発した不完全な言葉の行間を読みながら言わんとすることを理解する能力を人間が持っているからである。ここで、「行間を読む」という行為は、実にさまざまな人間の能力を結集した賜であるが、そのうちの一つとして、相手の発話中のある特定要素から適切な連想を行うという能力が挙げられる。したがって、談話理解も補助線問題の一種であると考えられる。本論文

で提唱した学習法を、これらの「一般の補助線問題」にも適用できるよう拡張することが、今後の課題となる。

謝 辞

この研究を進めるにあたり、(株)日立製作所基礎研究所の櫻井、吉田、山崎、本山氏らのアドバイスが我々の理解を深める大きな助けとなった。深く感謝します。

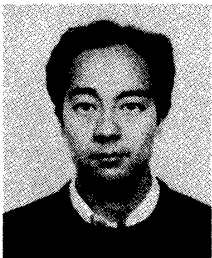
また、ICOT ワーキンググループ FAI での討論も非常に有益であった。委員諸氏、オブザーバーの方々に感謝致します。そして、主筆者(諏訪)にこの研究を基礎研究所で行う機会を与えてくださった基礎研究所丸山所長、東京大学工学部鈴木教授に深く感謝の意を表します。

◇ 参 考 文 献 ◇

- (1) Anderson, J. R. : The Architecture of Cognition. Chapter 4, Harvard University Press (1983).
- (2) Bain, W. M. : A Case-Based Reasoning System for Subjective Assessment, Proc. of AAAI-86, pp. 523-527 (1986).
- (3) Cheng, P. W. and Carbonell, J. G. : The FERMI System : Inducing Interactive Macro-Operators from Experience, Proc. of AAAI-86, pp. 490-495 (1986).
- (4) Dejong, G. and Mooney, R. : Explanation-Based Learning : An Alternative View, *Machine Learning*, Vol. 1, No. 2, pp. 145-176 (1986).
- (5) Goldstein, I. : Elementary Geometry Theorem Proving, AI Memo No. 280, MIT (1973).
- (6) Greeno, J. G. : A Study of Problem Solving, in R. Glaser (ed.), *Advances in Instructional Psychology*, Lawrence Erlbaum Associates (1978).
- (7) Hammond, K. J. : Learning to Anticipate and Avoid Planning Problems through the Explanation of Failures, Proc. of AAAI-86, pp. 556-560 (1986).
- (8) Laird, J. E., Newell, A. and Rosenbloom, P. S. : SOAR : An Architecture for General Intelligence, *Artif. Intell.* Vol. 33, pp. 1-64 (1987).
- (9) Lebowitz, M. : Integrated Learning : Controlling Explanation, *Cognitive Science*, Vol. 10, No. 2, pp. 219-240 (1986).
- (10) Minton, S., Carbonell, J. G., Etzioni, O., Knoblock, C. A. and Kuokka, D. R. : Acquiring Effective Search Control Rules : Explanation-Based Learning in the PRODIGY System, Proc. of the 4th International Workshop on MACHINE LEARNING, pp. 122-133 (1987).
- (11) Minton, S. : Selectively Generalizing Plans for Problem-Solving, Proc. of IJCAI-85, pp. 596-599 (1985).
- (12) Mitchell, T. M., Keller, R. M. and Kedar-Cabelli, S. T. : Explanation-Based Generalization : A Unifying View, *Machine Learning*, Vol. 1, No. 1, pp. 47-80 (1986).
- (13) Mitchell, T. M., Utgoff, P. E. and Banerji, R. : Learning by Experimentation : Acquiring and Refining Problem-Solving Heuristics, in R. S. Michalski, J. G. Carbonell, T. M. Mitchell (eds.), *Machine Learning : An Artificial Intelligence Approach Volume 2*, Morgan Kaufmann (1986).
- (14) Shavlik, J. W. and Dejong, G. F. : BAGGER : An EBL System that Extends and Generalizes Explanation. Proc. of AAAI-87, pp. 516-520 (1987).
- (15) Ullman, S. : A Model-Driven Geometry Theorem Prover, AI Memo No. 321, MIT (1975).
- (16) Wong, R. : Construction Heuristics for Geometry and A Vector Algebra Representation of Geometry, Memo No. 28, MIT (1972).
- (17) 諏訪正樹, 元田 浩 : フラストレーションに基づく学習, 人工知能学会研究会資料 SIG-FAI-8801 (1988).
- (18) Anzai, Y., Ishibashi, N., Mitsuya, Y. and Ura, S. : Knowledge-based problem solving by a labelled production system, Proc. of IJCAI-79, pp. 22-24 (1979).

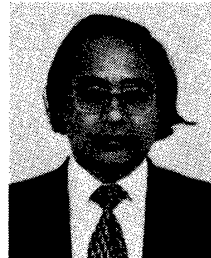
[担当編集委員・査読者：安西祐一郎]

著 者 紹 介



諏訪 正樹 (正会員)

1984年東京大学工学部原子力工学科卒業。1986年同大学院原子力工学専攻修士課程修了。現在、同博士課程在学中。人工知能、特に学習の研究に従事。人工知能学会、情報処理学会、日本認知科学会、日本原子力学会、AAAI各会員。



元田 浩 (正会員)

1965年東京大学工学部原子力工学科卒業。1967年同大学院原子力工学専攻修士課程修了。工学博士。同年、日立製作所に入社。同社中央研究所、原子力研究所、エネルギー研究所を経て、現在、基礎研究所主管研究員。原子力システムの設計、運用、制御に関する研究。診断型エキスパートシステムの研究を経て、現在は人工知能の基礎研究、とくに推論機構に関する研究(定性推論、学習、知識コンパイルなど)に従事。1970年日本原子力学会奨励賞、1977年日本原子力学会論文賞、1984年日本原子力学会論文賞受賞。人工知能学会、情報処理学会、日本ソフトウェア科学会、日本認知科学会、日本原子力学会、AAAI、ANS各会員。