

図を用いた問題解決を支援するための学習環境について

左右田 東虎 柏原 昭博 平嶋 宗 豊田 順一
大阪大学産業科学研究所

CAIが提供する学習環境の下で学習者が効果的に学習を行なうためには、学習者の状態に対応して環境を制御する必要がある。環境の制御とは、環境が学習者に与える制約を緩和あるいは強化することである。本研究では、学習における負荷の概念を導入し、学習者に対して常に適度な負荷をかけ続けるように環境の制約を制御していく。学習者にかかる負荷としては、インタフェースの提供する学習ツールによるもの、およびチュータの介入によるもの、対話相手によるものを挙げる。学習者に適度な負荷をかけることによって、学習者の学習意欲を高め、かつ問題理解を深めることができると考えられる。本稿では負荷制御による環境制御を実現するために、最大値、最小値を対象とした学習環境を作成した。

Learning Environment for Problem Solving with Diagram

SO'OTA Akira KASHIHARA Akihiro HIRASIMA Tukasa TOYODA Jun'iti

The Institute of Scientific and Industrial Research, Osaka University

This paper describes learning environment in CAI with environment control. In order for a student to learn freely in the environment, CAI system must appropriately control the environment. The environment control is to lighten or to strengthen several constraints embedded in the environment. This research introduces the load applied to the student's learning for environment control. Our system controls the constraints in the environment to continuously apply the adaptive learning load to a student. We have classified the load into three types: learning tools in interface, tutor's interrupt, and fellow student's proposition. The adaptive load control contributes to the promotion of the student's learning motivation and his/her problem understanding. Based on this approach, we have implemented a learning environment for problem solving with diagram.

1. はじめに

ITSを中心とするこれまでのCAIシステムでは、学習者の思考における負荷をできるだけ軽減する支援機能の構築に力点が置かれてきた。たとえば、わかりやすく説明を行なう機能や適切なヒント、例題を提示する機能などがある。負荷の軽減は、計算機以外の教育媒体(教科書など)に対するCAIの優位性を示す上で重要な観点である。しかしながら、負荷の軽減を指向した場合、学習者が受け身になりやすい、学習の達成感が得られにくいなどの問題点がある。これらは、知識伝達をより高度化するためには見逃せない重要な問題である。筆者らは、学習における負荷を軽減するだけでなく、学習者の学習能力に応じて動的に負荷の程度を制御する計算機メカニズムの実現を目指して研究を進めている[柏原92]。

負荷の軽減による問題点を解消するためには、学習者の学習に対して故意に負荷を与えることが有効であると考えられる。この場合、負荷が大きすぎると学習意欲を損ねたり、逆に小さすぎると学習の達成感が得られないといった恐れがある。しかしながら、負荷が適切に設定されれば学習者の注意を喚起することができ、学習意欲を高揚させることができる。また、学習者は負荷を与えられると学習に行き詰まることが想定される。このような行き詰まり(Im-passe)の状況を積極的に作り出し、システムがそれを打開するような支援を学習者に与えることによって、負荷の軽減を指向した場合よりも学習者による学習を強化することができる。これらの長所を活かすためには、うまく負荷の程度を制御できなければならない。本稿では、教育支援システムが提供する学習環境における制約(環境制約と呼ぶ)を緩和、強化することによって負荷制御を行う手法について考察する。

学習者は、教育システムを用いて学習活動を展開する場合、システムから様々な制約を受ける。例えば、システムからの説明や練習問題の提示によって学習者の思考は束縛を受ける。また、環境型CAI[林92]やマイクロワールドでは、あらかじめ用意されたツールそのものが学習者の学習活動を制限することになる。このような環境制約をシステム運用時に

緩和、強化することによって、学習の負荷を変化させることができる。直感的には、学習者に与えられる学習環境を動的に変化させることによって負荷の制御を行おうとするものである。本稿では、環境制約を(1)学習者が操作可能なツール、(2)チュータの介入度、(3)対話相手、の三つに分けて捉えており、これらの緩和、強化による負荷制御について検討する。

以下、2.では環境制約と学習の負荷の関係について議論する。3.、4.では、2.での議論をもとに現在開発を進めている本システムについて述べる。

2. 環境に対する考察

2.1. 環境の制約

本節では、(1)学習者が操作可能なツール群、および(2)チュータの介入、(3)対話相手、の3つの点から環境制約について、負荷との関係を考慮しながら考察する。

(1) 学習者が操作可能なツール群

教育システムを用いて学習を進める場合、一般にはシステムのインタフェイス上に様々な学習用のツールが用意される。学習者はこれらのツールを用いて学習を進めることになる。通常、システムに埋め込まれた問題解決モデルにそって正解を導くツールだけが用意されている。このとき、学習者の思考は、その問題解決モデル内に制限されてしまうが、あまり負荷がかからずに学習および問題解決が行なわれる。一方、この問題解決モデルに直接そわないような外乱となるツールをインタフェイス上に用意しておけば、学習者の学習行動は幅広くなる。学習行動の選択肢が幅広くなることによって学習者の問題解決に思考錯誤が発生するため、学習に負荷がかかると考えることができる。操作可能なツールを増やすことは環境制約を緩和することに対応し、ツールを減らすことは制約を強化することに対応する。このようにみれば、制約の緩和および強化は、学習における負荷の増加および減少につながる。

また、一般には同じツールでも、学習者の学習行動を促進する側面と、学習行動を阻害する側面をあわせもっている。このようなツールの効果を意識的に強調することによって、学習の負荷を制御すること

ができる。(具体的な例は4.で挙げる)

(2) チュータの介入によるもの

学習者が学習を進めていく段階でチュータが積極的に介入すると、学習者の思考は強制的に狭められていくと考えることができる。逆に介入の度合いが少ないと、学習者の学習行動は幅広くなる。その代わりに、それに見合う学習の負荷を負うことになる。このように、チュータが介入する頻度は学習者への負荷の要因となる。つまり、チュータの介入頻度が高いと制約が強化されるので負荷は減少し、介入頻度が低いと制約が緩和されるので負荷は増加する。

(3) 対話相手

通常の学習環境では、学習者の対話相手は教育システムに埋め込まれたチュータである。本研究では学習の負荷制御といった観点からチュータ以外に fellow student を対話相手として想定する。fellow student とは学習者と同等の知識レベルにあり、学習者が現在取り組んでいる問題に対して fellow student 自身の意見を述べる。fellow student の提示する解は正しいとは限らず、多くの場合、いくらかの誤りを含んでいる。このような fellow student を対話相手とすると、ある場面では学習者は fellow student を相手に教授行動 (teaching) をとることになり、またある場面では fellow student から誤った知識を教えられたりすることになる。そのため、学習者はチュータを対話相手にする場合よりも、学習行動に対して多くの注意を払う必要があり、学習の負荷は大きくなると予想することができる。

なお、学習者が fellow student の誤った学習や問題解決に賛同した場合には、チュータが何らかの形で介入し、教育活動を行うことを想定している。

2.2. 負荷制御

以上のような環境制約をむやみに大きく緩和すると、学習者の学習意欲 (motivation) が失われてしまう危険性がある。また、強化しすぎても同様に学習意欲をそぐ恐れがある。学習環境内において学習意欲を持続させるためには、学習者の能力に従ってシステムが適度な負荷を動的に与え続けることが効果的であると考えられる。学習者にかかる負荷がある程度大きくなっていくと学習者の学習意欲は高まって

いき、負荷が大きくなりすぎると学習意欲は低下すると考えられる。逆に、負荷が小さくなりすぎると、学習の達成感が減少することが予想されるので、学習意欲も低下すると考えられる。学習者の学習意欲を高める適度な負荷領域があるはずである。学習者に適度な負荷をかけるために、学習環境は、負荷制御のための機能を備えなければならない。本研究では、現在、負荷制御を行う手段として、fellow student の問題解決能力の制御およびインタフェース上に用意するツール群の制御の2つをとり上げている。

2.2.1. fellow student との対話

fellow student (学友)とは、問題解決モデルをもとにしたいくつかの戦略から様々な解答を得ることができる問題解決器であり、バグモデルから誤りを再現するものである。さらに、fellow student は学習者との対話を行なうためのプロシージャを持っている。fellow student が生成する解の誤り度はシステムが制御する。誤った解を生成する fellow student を学習者と対話させることによって学習者に負荷をかけることができる。対話プロシージャには、主張モードおよび、質問モード、学習モードの3種類のモードがある。主張モードでは、fellow student は自分の意見を積極的に述べ、自分の主張が正しくなるような例などを提示する。質問モードでは、学習者の行なった学習および問題解決に対して疑問点を指摘する。学習モードでは、聴き手にまわり積極的な発言をせず、学習者の説明から学習を行なう。

このような fellow student を作成し有効に活用するためには、(1) 問題解決モデルをもとにした誤りの抽出と整理、および (2) (1) で作成したバグモデルを正当化する問題の条件の抽出と整理、(3) 学習者からの説明にもとづいて学習を行なうメカニズムの構築、を行なう必要がある。本研究で開発したシステムでは、(1)、(2) についてのみ行なった。

2.2.2. ツール群の変更

前述のように、ツールの数を増やすことは、学習者にかかる負荷を大きくすることになる。大きな負荷に耐えることができる学習者、すなわち学習能力が高い学習者に対しては、学習環境内で可能なすべての操作をツールとして与えることによって、学習行

動への制約を緩和する。逆に、学習能力の低い学習者に対しては、できるだけ正解に導くようなツールだけを選択して用意し、なるべく負荷をかけないようにする。このようなツールの制御を行なうためには、システム設計時において設計者が想定する問題解決モデルに沿って、正解に導くツールや直接モデルにそぐわないツールを整理しておくかなければならない。

また、ツールが有する正負の効果を予め考慮すれば、ある場面では正の効果を強調することによって学習を促進させたり、逆に負の効果を強調することによって、学習の負荷をかけることができる。ここで、強調するとはツールによって引き起こされる正負の効果をインタフェースで強く可視化することに相当する。このように、ツールによる負荷制御を行なうためには、ツールが有する正負の効果を抽出しておく必要がある。

3. 問題解決および誤りのモデル化 [左右田 92]

本章では、前章までの考慮にもとづいて開発を進めているシステムについて述べる。特に、扱っているドメイン、および想定している問題解決モデル、誤りのモデルについて述べる。

3.1. 問題領域の設定

高校数学の問題には、最大値、最小値や領域問題、軌跡問題などのように数式と図形との対応関係によって解決することができるものがある。本研究では、最大値、最小値問題を対象として、学習者が問題解決のための知識を学習する学習環境の設計、開発を目的としている。最大値、最小値問題は、数式のみを用いて解くことが可能である。しかれども、数学においては図を描きながら考えることが重要な働きをする場合が多い。これを受けて、数式と図形との対応づけを速やかに行うことができるように学習者の学習を支援することが筆者らの狙いである。

以下では、問題解決過程のモデル化について考察

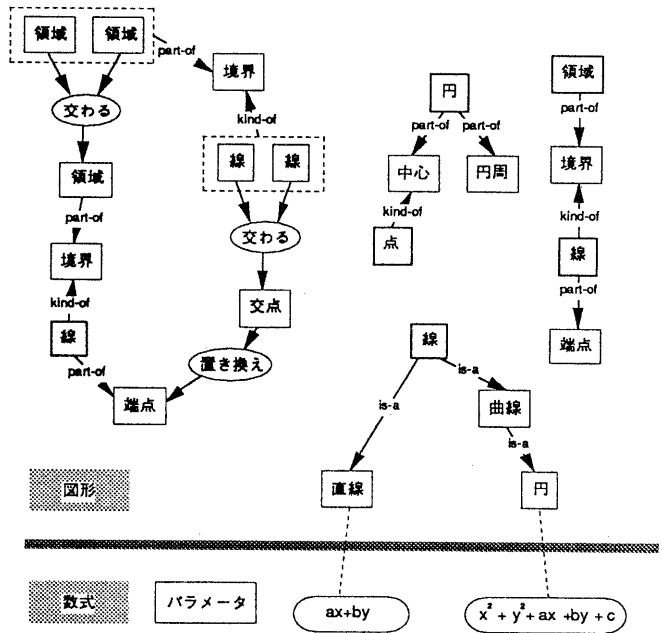


図 1 領域概念

し、それに基づいて予想される学習者の誤りについて述べる。

3.1.1. 領域概念構造

最大値、最小値問題の領域では、数式と図とを対応づけて問題解決が行われる。ここでは、まず問題解決過程を捉えるために、最大値、最小値問題で用いられる概念を抽出し、それをを用いて問題解決モデルを表現する。図 1 に最大値、最小値問題で用いる領域概念構造の一部を示す。問題解決に用いる概念は、図形に関する概念構造と数式に関する概念構造に大別することができる。これらは xy 座標系によって対応づけることができる。概念構造における概念間の関係には part-of 関係および等価関係がある。領域概念間の関係には、さらに is-a 関係および kind-of 関係がある。

3.1.2. 問題解決モデル

最大値、最小値問題の問題解決モデルは図 2 のようになる。問題解決は数式表現と図形表現に分けることができる。図 2 によれば、数式表現では、問題解決を「制約条件をみたまうように数式のパラメータを変化させる」ことであると表現することができ

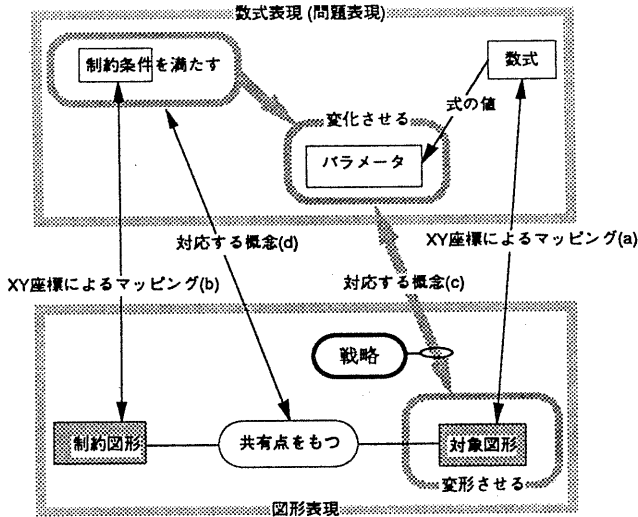
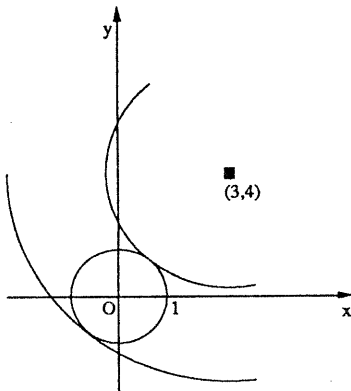


図 2 問題解決モデル

る。例えば、図 3 例題 1 では、制約条件とは $x^2+y^2=1$ であり、数式とは $x^2+y^2-6x-8y = (x-3)^2+(y-4)^2 = k+25 = r^2$ である。パラメータは r を意味する。図形を用いた問題解決は、数式表現と図形表現との対応付けおよび図形表現上での解決によって表現される。すなわち、(a) パラメータを求める数式を図形(対象図形と呼ぶ)に変換し、(b) 制約条件を図形(制約図形と呼ぶ)に変換する。さらに、(c) パラメータの変化

[例題 1]

実数 x, y が $x^2+y^2=1$ をみたしながら変化するとき $x^2+y^2-6x-8y$ の最大値と最小値を求めよ。



[例題 2]

x, y が連立不等式 $x-3y \geq -6, x+2y \geq 4, 3x+y \leq 12$ を満たすとき x^2+y^2 の最大値と最小値を求めよ。

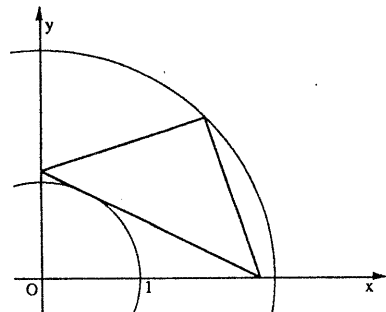


図 3 例題

を対象図形の変形として捉え、(d) 制約条件を満たすことを、変形した対象図形が制約図形と共有点をもつことであると捉える。本研究では、このような 4 つの段階によって問題解決過程を捉えている。例えば、先の例における問題解決は、次のように表現される。つまり、まず制約図形として原点を中心とする半径 1 の円を、対象図形として $(3, 4)$ を中心とする半径 r (パラメータ) とする円をそれぞれ描く。次に、半径 r を大きくあるいは小さくすることによって、制約図形と共有点を持つ r の範囲を求める。最終的には、その範囲における r の最大値、最小値が求められる解となる。

このように、図形を用いて行う問題解決は、「制約図形と対象図形とが共有点をもつようにパラメータを変化させる」ことであると表現することができる。図 2 における共有点や制約図形・対象図形といった概念の部分は、個々の具体的な問題に応じて、図 1 で整理した具体的な概念がインスタンス化される。

3.2. 問題解決のための戦略

図 2 の問題解決モデルにおいては、対象図形をど

のように変形させるかが問題解決に大きな影響を与える。対象図形を変形していく場合、図形の動かす方向と着目点(動かす範囲)が重要となる。この2つを併せたものを戦略とする。この戦略は、対象図形および制約図形のタイプによって異なったものとなる。本稿では、制約図形を直線および円、円弧、三角形の領域とし、対象図形では直線および円に限定している。このような前提のもとに、制約図形と対象図形の組み合わせごとに戦略を整理した。その結果を表1に示す。たとえば、図3例題1のように対象図

表1 対象図形と制約図形の組み合わせによる戦略

対象図形	制約図形	対象図形の動かし方	範囲を限定するための着目点
直線	円	平行移動	接点 - 接点
直線	円	傾きの変化	接点 - 接点
直線	円弧	平行移動	接点 - 端点
直線	円弧	傾きの変化	接点 - 端点
円	円	半径の変化	接点 - 接点
円	三角形	半径の変化	半径を拡大する
円	三角形	半径の変化	半径を縮小する
円	三角形	—————	端点 - 端点

形も制約図形もともに円である場合には、対象図形の動かし方は半径を変化させることで、制約条件を満たすような半径の範囲は、対象図形の円と制約図形の円との内接点から外接点までである。

3.3. 問題解決における行き詰まりと誤り

本節では問題解決モデルにもとづいて予想される行き詰まりと誤りを分類する。

3.3.1. 行き詰まりの分類

現在のところ問題解決に対応する行き詰まりの原因を以下のように分類している。

- (a) 数式と図形との対応づけができない
- (b) 制約条件と集合との対応づけができない
- (c) パラメータの変化と図形の変形との関連づけができない
- (d) 制約条件を満たすことと共有点をもつことが等価になることがわからない

以下では、個々の行き詰まりの例を示す。

(a) 数式と図形の変換を行うことができない、あるいは数式の変形だけで解を求めようとする場合である。例題1の場合、 $x^2+y^2=1$ より $x=\cos\theta$ 、 $y=\sin\theta$ であるから、これらを関数に代入して計算により最大値、最小値を求めることができる。このこと自体は誤りではないが、式と図形とを対応づけることに行

き詰まっていると本研究では考える。この行き詰まりは、数式に対応する図形概念が形成されていないことが原因であると考えられる。

(b) 制約条件を表す図を描くことができない場合である。例題2の場合、 $x-3y\geq-6$ 、 $x+2y\geq4$ 、 $3x+y\leq12$ を満たす領域を図示することができない。この原因としては、(a)と同様に数式と図形を表す概念のマッピングがとれていないことが考えられる。

(c) 図3例題1では、関数が点(3,4)を中心とする円を表すことがわかっていながら、パラメータkを変化させることがこの円の半径を変化させることに気づかない場合である。本来は、パラメータkによって半径が変化する円として、この円を捉えなければならない。

(d) 図3例題1では、制約条件となる円を描くことができ、関数が表す円はパラメータkによって半径が変化するがわかっていながら、2つの円の関係がどのようになっていれば題意を満たすことができるのかわからないために、行き詰まってしまう場合である。

3.3.2. 問題解決の際に発生する誤り

学習者の誤りは、問題解決の各段階で発生すると考えられる。本稿では、特に対象図形を変形して制約

図形との共有点を見つめる段階での誤りについて考察する。すなわち、対象図形を变形する戦略の誤り、特に戦略適用の誤りに限定して議論する。

前述のように、戦略には対象図形の動かし方と動かす範囲の着目点が含まれている。戦略適用の誤りには、(1) 動かし方が原因となる誤り、および(2) 着目点の設定の誤りがある。本稿では対象図形が円で制約図形が三角形である問題（これを円-三角形問題と呼ぶ）における誤りについて述べる。円-三角形問題で発生する

誤りの例を図4に示す。(1)は、対象図形である円の動かし方そのものは誤りではないが、半径を小さい方向から大きい方向へ動かすために最大となる共有点を誤る例である。この動かし方では、最小値は正しく求められるが、共有点が三角形の辺ABとの接点であるため最大値も同様に辺BCとの接点と考えられ、誤りが誘発されやすくなるといえる。また、(2)は着目点の設定の誤りであり、図を動かすことを考慮せず三角形の頂点を共有点の最大値・最小値となる点として見なす（端点戦略と呼ぶ）ために発生する誤りである。

4. システムについて [左右田 92]

3.で整理したモデルをもとに、2.での主張を踏まえて、学習環境を実現するシステムの作成を行なった。システムの概要は図5のようにになっている。シ

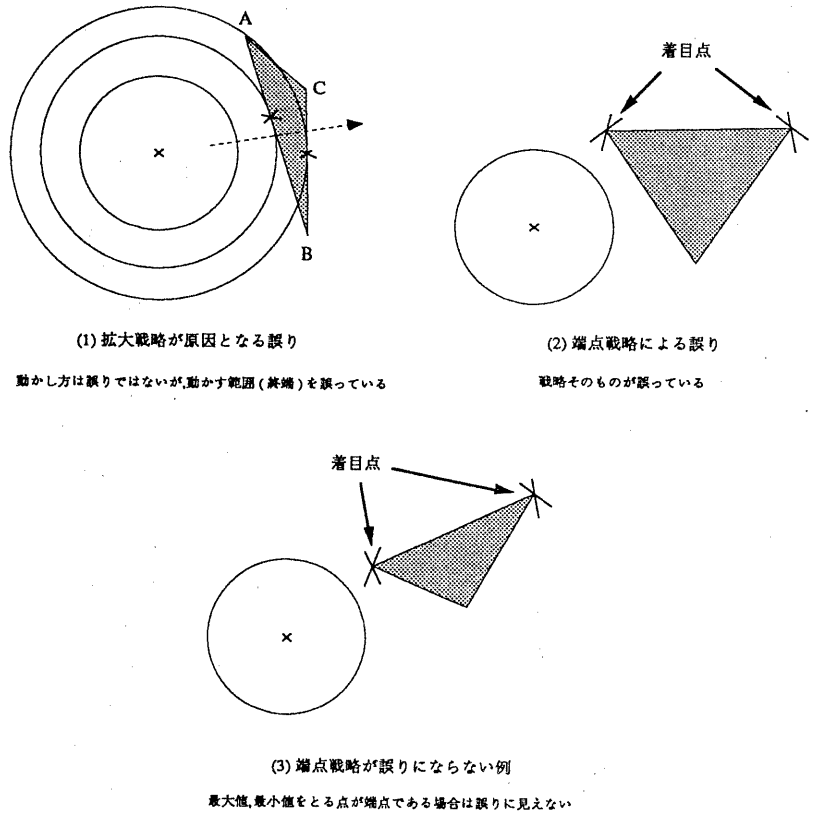


図4 問題解決戦略の例

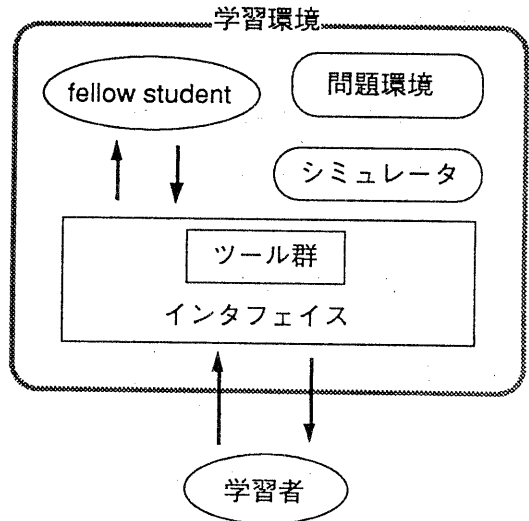


図5 作成したシステムの概要

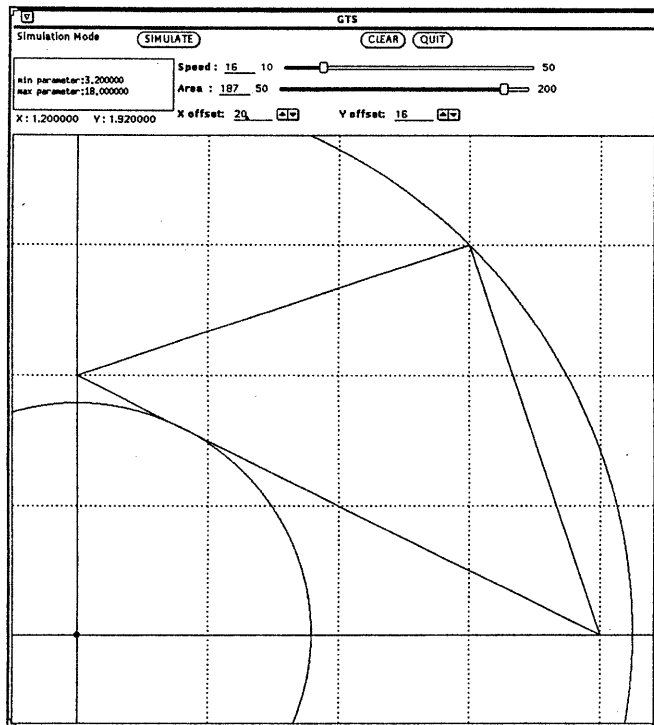


図 6 システムの様子

F: A点とB点に着目すればいいんじゃないか?

なんかおかしいな

S: 距離が最小になるのはA点かな?

F: そうだよ

S: この点の方が距離が短いとおもうんだけどね

F: そうかな? でも考え方は正しいはずだよ.

S: どうして?

F: 同じようなこの問題(図3(3))で,うまくいくでしょ

なぜだろう

S: 確かにうまくいっているな. でもねえ.

F fellow student

S student

図 7 対話例

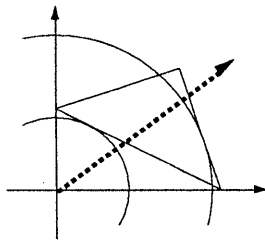
ミュレータは,学習者や fellow student が行なった問題解決を画面以上に実行する装置である. 現在のシステムでは, (A) 制約図形と対象図形が共に円であるもの, および (B) 制約図形が三角形の領域で対象図形が円であるものの2種類を扱っている. 図6に対象図形を変形する実際の様子を示す. インタフェイス上には, 対象図形や制約図形を描くツールや対象図形を動かすためのツールが用意されている. fellow student は, 3. で述べた対象図形の変形戦略を誤った形で問題解決を実行したり, 学習者との対話を行なう対話プロシーダを有している. 本システムは, SUN SPARCstation2 上の日本語 OpenWindows を用いてすでに実現されている.

本システムでは, 現在のところ fellow student との対話および問題解決に用いるツール群の用意の仕方によって学習の負荷を増減させることを可能としているが, きめ細かな負荷制御は実現されていない. 以下では, 本システムにおける学習

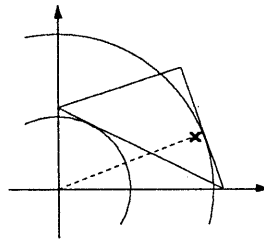
への負荷のかけ方について述べる.

4.1. fellow student による負荷

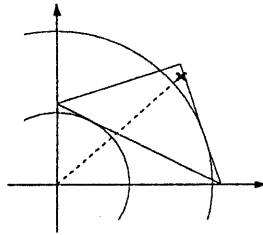
fellow student と学習者の対話には, fellow student による主張と学習者への質問の二つのモードがある. 主張モードでは, fellow student が対象図形の変形についてあるタイプの戦略をいくつかの問題に対して適用しながら最大値, 最小値を求める. このとき, その戦略を用いて正しく解が求められる問題と求められない問題が選択される. fellow student の問題解決プロセスは, 学習者に提示される. 学習者は, fellow student が行なった問題解決に対して誤っているあるいは正しいなどの意見を述べるができる. 例えば, 端点戦略によって正しく解ける問題(図3(3))と正しく解けない問題(図3例題2)を fellow student が解決している場面での対話例を図7に示す. 学習者が fellow student の誤った問題解決に賛同した場合には, チュータの介入を行なうことを想定している. このような fellow student との対話は, 従



(1) 拡大戦略による最大値の誤り



(2) 円からコンパスへの置き換え



(3) 更に外側への移動

図 8 ツールの置き換え

来のチュータのみの対話に比べて、学習者はより多くの注意を学習に払う必要があるため負荷がかかると考えられる。

現在, fellow student が行なう戦略は, 端点戦略による誤りを含むもの, および, 拡大戦略で接点を最大とする誤りを含むもの, 拡大戦略を正しく行なうもの, 縮小戦略を正しく行なうものの4つに分けられる。端点戦略は誤った戦略であるが, 拡大戦略で最大

値を誤るのは, 戦略自体の誤りでなく, 戦略適用の誤りである。

fellow student との対話モードは, 現在学習者が自由に選ぶこととしており, 対話を望めば fellow student の行動が割り込んでくる。また, 対話をしないようにしていても, システムの判断により, fellow student との対話の機会を作ることも可能である。

4.2. 学習環境のツール

現在, 対象図形や制約図形を描くツールと対象図形を動かすためのツールが用意されている。学習者は図形を描くツールを用いて問題から与えられた図形を xy 軸上に描くことになる。具体的には, 座標空間へ貼込む図形をメニューから選択することになっている。さ

らに, 対象図形を動かすツールによってシミュレーションを実行する。このとき, 学習者は, 動かす対象を選び (例: 円の半径) 動かす向きを指定することになる。また, 指定した点が最大値, もしくは最小値であると宣言する方法もある。これらの操作は, 学習者が対象図形を変形する戦略を言明していることを表現している。言明された戦略は, シミュレータに渡されてシミュレーションが実行される。

本システムでは, 図形を描く際にメニューとして用意する図形のタイプ (円や三角形など) を制約したり故意に問題と関係のないものを用意したりすることによって, 学習の負荷の程度を制御することを試みている。このようなメニューの変更は, 図 2 に示した問題解決モデルにおける数式表現から図形表現にマッピングする際の負荷を制御しようとするものである。

また, 座標空間上にある図形を, 学習者はマウスを用いて掴み図形を動

表 2 式 $x^2 + y^2 - ax - by$ を表現する対象図形の機能

円

正	距離の違いや変化がわかりやすい
負	不要な部分まで意識してしまう

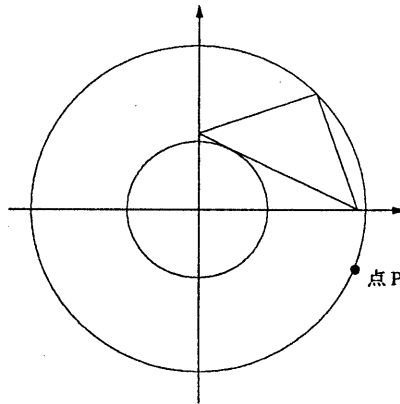
コンパス

正	本質的な距離だけを認識しやすい
負	距離の変化がわかりやすい

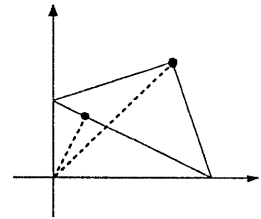
かすこともできる。例えば、問題解決に直接関係のない「円の中心を移動させる」といった操作を行なうことも可能である。このような操作を許すことは、対象図形と制約図形の共有点を見つけようとするプロセスに対する負荷の程度を制御することにつながる。

また、学習者は対象図形を变形させるツールを自由に選ぶことができる。例を挙げると、図3例題1、例題2の対象図形を、円とコンパスの2つのツールによって变形することができる。この2つのツールは、それぞれ正と負の効果を持つ。表2にそれらの効果を示す。

円またはコンパスを用いた場合、それぞれどのような効果があるか説明する。例題2において、学習者が、円を用いて拡大戦略のシミュレーションを行なった時に、図8(1)のように最大となる半径を誤った場合を考えてしまう。システムが円をコンパスに置き換えれば図8(3)のように、点を外側へ移動させることが可能であるが円を用いた場合よりも顕著に理解することができる。また、最小、最大となるパラメータに対応する対象図形として、図9(1)のように円を描いておいた場合に、図中の点Pは何を意味するのかといった疑問のために、問題解決に行き詰まってしまうことが予想される。ここで図9(2)のようにコンパスモデルを用いて結果を示せば、最大値を取る点を明示的に示すことができるので、前述のような疑問は起こらない。この場合、コンパスモデルを用いて、シミュレーションを行なえば、点の所在と動きはわかりやすくなるが、逆に距離(パラメータ)の変化がわかりにくくなる。これらのツールによる効果を考慮したツールの用意の仕方を行なえば、問題解決において学習者にかかる負荷をうまく扱うことができると考えられる。



(1) 円による結果表示



(2) コンパスによる結果表示

図9 ツールによる効果

5. まとめ

本稿では、CAIにおける学習環境に負荷制御の考え方を導入することによって学習者の学習意欲、学習の強化が行なえるという提案を行なった。また、提案した学習環境を実現するために、図を用いることによる最大値、最小値問題の問題解決を支援するシステムを実現した。今後の課題は、本システムにおいて、さらにきめ細かな負荷制御を実現することが挙げられる。

謝辞

本稿の図表作成にご協力いただいた、大阪大学大学院工学研究科応用物理学専攻博士課程前期(修士課程)1年 松村 浩一君に感謝します。

参考文献

- [柏原92] 柏原他: "知識伝達失敗へのアプローチ-リアクティブな説明プランニングの枠組みについて", CAI学会第17回全国大会講演論文集, pp. 291-294 (1992).
- [左右田92] 左右田他: "説明図を用いた問題解決支援のための学習環境の枠組み", CAI学会第17回全国大会講演論文集, pp. 283-286 (1992).
- [林92] 林他: "漢字学習を対象とした環境型知的CAI", 情処研報, 92-CE-23, pp. 57-66 (1992).