コンピュータと教育　19−14
（１９９１．１１．１６）

知識ベースに制御されたメディアインターフェイスによる学習支援

伊藤紘二　伊丹　誠
東京理科大学基礎工学部

本稿は、学習支援環境ＣＡＦＥＫＳにおいて、メディアを用いたメタファ表現を導入する試みについて提案している。これによれば、システムの支援によって学習者は、問題の対象世界の認識を知覚・運動感覚に訴えるメディアによって表現することができ、システムは、知識の適用や問題解決のコンテキストをメタファによって提示する。学習者は、これらの表現に介入して結果を見ることができる。実現には、Ｘウインドウ環境において管理実行されるメディアインタフェースオブジェクトを、Ｃ言語の上のオブジェクト指向で構成し、Prologで書かれる知識ベースのクラスに置かれたメソッドから、そのクラスのメディアを制御する。なお、いくつかの分野で試作中のオブジェクトの仕様について論じている。

ASSISTING LEARNERS BY MEANS OF MEDIA INTERFACE OBJECTS
CONTROLLED BY KNOWLEDGE BASE OBJECTS

Kohji Itoh　Makoto Itami
Science University of Tokyo, Noda 278 Japan

The present paper proposes incorporating into our problem solver's assistant such methods as assist the student interactively to create instances of the problem object world in static or dynamic media presentation and to interact with the presentation. Also proposed is presentation of knowledge applications as well as the problem solving progress and contexts. In implementation, media interface object classes are programmed in C, managed and executed in X-window environments. Media control methods in the knowledg-base classes programmed in prolog send messages to the media interface objects via inter-process communications, thus activating media presentation and accepting user intervention. Some of the interim specifications of such methods are described in the context of several domains in which prototyping is now under way.

1．まえがき

われわれは、学習者の問題解決を支援するコンサルタント型のシステムーＣＡＦＥＫＳーを試作している[1][2]が、多くの領域において、自然言語を含む分節的記号を用いたインターフェイスだけでは、到底ヒューマンな支援は不可能であることを認識した。

結局、知覚や運動感覚を通して獲得され環境に直結した認知の世界に、我々の思考は、その根拠をおいているのであり、意識は、直接その仕組を覗くことは最早できない[3]。

従って、問題解決過程で現れる認識や知識や問題について、この世界に直接訴え掛ける仕組なしには、ユーザとの対話に非常な困難が生ずる。

しかしこのことは、いわゆる仮想現実の世界の必要を意味しない。むしろ、抽象的な事柄をも、適切なメタファによって、知覚的、空間操作的に表現してやることが重要なのである[4]。

本稿では、ＣＡＦＥＫＳにおいて、メディアを用いたメタファを導入する試みについて提案する。まず、ユーザとの対話の媒介として、あるいは静的なあるいは動的なメディア表現とその上におけるユーザの介入の受け付けを行うメディアインターフェイスオブジェクトを定義する。そして、学習支援において不可欠な知識ベースを構成する認識、知識、問題およびそのサブクラスにこれらのメディアインターフェイスを制御するメッセージ送信手続きを置く。知識ベースクラスのインスタンスおよび属性値は、この仕組を用いることにより、学習者あるいはシステムによってメディア表現される。またユーザは、この表現のなかでパラメータを操作して結果を見ることが出来る。

続いて、現在進行中のいくつかの試作事例について、構想を述べるが、この仕組により、問題解決と知識獲得の諸場面における対話が容易になり、学習者による方略や知識の発見を支援できることが期待される。

2．学習支援における知識ベースとメディア表現

G.Polya[5]によれば、問題解決のサイクルは、問題の理解、解決の計画、（副問題）解決の実行、結果の吟味の４つのフェーズからなり、その再帰的ないし、やり直し的プロセスを辿る。ＣＡＦＥＫＳはいずれのフェーズをも支援する。

まず、問題の理解のフェーズとして、システムは問題型のクラスから、学習者に問題の記述枠を提供しながら、認識型のクラスによって、対象世界の記述枠を提供しつつ、対話的に問題を記述させる。この行為によって学習者は問題を整理して観察し、解決の方略についての仮説をたてることを期待される。このプロセスは、問題解決のサイクルのなかで最も非分節記号的な思考の役割が大きい部分である。従って、認識型の諸クラスのインスタンスの生成と修正、あるいは介入操作の出来るシミュレーションを、知覚・運動感覚に訴えるメディア表現の上で行えることが極めて重要である。

解決の計画のフェーズとして、システムは、方略の候補を提供する。これは、問題解決手順と副問題のならびからなる解決計画ないしヒュリスティックスであり、さらに副問題の型を検索することによってその方略を知り、こうして、問題全体の解決の方針を予め立てることができる。これは、かなり抽象化されたプロセスなので、例題を検索させて理解を助けることが必要である。従って、例題の問題記述と問題解決過程（後述）ならびに結果をメディア表現で提供することが必要である。

解決の実行のフェーズでは、選ばれた方略に沿って、使えそうな知識の検索と適用を支援する。知識の検索においては、知識自体にデフォールト的な適用の提示と学習者の操作によって適用の仕方を変更できる機構を必要とする。知識の適用においては、その条件や結果を、問題コンテキストによるインスタンス化によってメディア表現する機構を必要とする。また、こうした対象世界におけるメディア表現と平行して、問題解決過程自体の進行状況をメタファで表わすことが重要である。そのために、関係／知識を節点で、変数／述語表現を枝で表わして節点を接続したグラフ、関数関係の表現などを用いる。

解の吟味は、問題解決過程のメディア表現による再現を必要とし、これを可能にするような管理機構が必要である。問題解決過程において生成されたメディアインスタンスの記録を保持し、これを利用して実現することができる。また、問題の一部変更がどのような結果を生ずるかの検討は、結果と解決手順の通用範囲を調べ、将来の問題解決の為に役立てる上で重要であるが、手順は知られているので、メディアの支援があれば、速やかな試行が可能である。

この様な仕組を次の方法で実現している。メディアインタフェースオブジェクトについては、Xウインドウ環境で、C言語を用いたオブジェクト指向によってプログラムし、Prologで書かれた知識ベースオブジェクトに、メディアインターフェイスオブジェクトに対するメッセージ送受信の手続きをおき、それを介したプロセス間通信によってメディアインターフェイスオブジェクトとのやりとりを行う。

3．移動に関する文章題の例

我々は、試作した汎用の問題解決支援機構の試験の為に、複数の移動の組合わせからなる移動に関する文章題の知識ベースを用いた[2]が、問題記述、知識提示、適用のいずれの場面でも、記号とことばだけによるやりとりは、極めて frustrating なものであった。

そこで、道に関する認識型クラスには、グラフのノードとアークとして道を描かせるべくユーザを支援する手続き、またルートクラスには、こうして描かれた道を引用してルートを指定させるべく支援する手続き、移動クラスには、こうして指定されたルートに、そこを移動するときのパラメータを入力させ提示する、といった手続きを付加することを試みている。

方略に従って検索してきた知識を適用するフェーズでは、知識の内容を、メディア表現の上に、それに合せてインスタンス化して提示する仕組を知識型に付加する。

また、等式関係を接点とし、変数を枝とするグラフを、等式知識が見出される度に接続して作ってゆくことで、式の過不足や消去の方針を把握させることができる。

4．電気回路理論の例

電気回路についてのメディアインタフェースは、ユーザが回路を構成してゆくのを視覚的に支援する手続きと、瞬時における各節点における電流の分流、ならびに回路に沿っての電位の降下の表示を主な内容とする。

また、任意の重ねあわせをこの上で表現してやることにより、テブナン、ノートン、補償などの諸定理を説明することが出来る。

抵抗回路を解く問題の解決支援には、閉路方程式、節点方程式による方法[6]と、拘束伝搬による方法[7]が考えられるが、いずれも、等式関係を節点とし変数を枝においたグラフを、方程式が見つかるごとに、独立性のチェックをしつつ、接続して提示してゆくことで、ユーザが方針を立てるのを支援することができよう。

また、ユーザのデザインした任意の集中定数線形回路において、過渡現象の解析を支援しながら諸現象の説明を生成するシステムを目指している。このシステムでは、全てのキャパシタの部分を電圧E_{Cn}の電圧源とし、全てのインダクタの部分を電流I_{Lm}の電流源とし、こうしてできる抵抗回路網の解析から、各電源から回路の任意の部分の電圧／電流へのトランスミッタンス（および自身から見たイミッタンス）を、求めておく。

このトランスミッタンス（とイミッタンス）を用いて、初期条件に対する回路の任意部分における初期電圧、初期電流の説明が生成できる。

さらに、これらの電源、電流源のそれぞれの電流i_n、電圧v_mについて、

$C_n \cdot dE_{Cn}/dt$

$= i_n(\{E_{Cn}, I_{Lm}\}$、外部電圧電流源）（１）

$L_m \cdot dI_{Lm}/dt$

$= v_m(\{E_{Cn}, I_{Lm}\}$、外部電圧電流源）（２）

とすれば、外部電圧電流源に駆動されるほかは、E_{Cn} I_{Lm}だけに関する一階連立線形常微分方程式を得る。

ここで外部電流源がないかあるいは直流電源であるとき、（１）（２）の左辺を０とおいて、E_{Cn}、I_{Lm}について解けば、充分時間が経った時のキャパシタ、インダクタの電圧電流が求まる。これは、各電源位置での他の電源からのトランスミッタンスとその位置からのイミッタンスを用いて、説明が出来る。

さらに、（１）（２）式において、対象世界状況として、$\{E_{Cn}, I_{Lm}\}$を与えれば、Δt後の$\{E_{Cn}, I_{Lm}\}$の状況が、ただちに与えられる。しかも、ΔE_{Cn}、ΔI_{Lm}を、一つづつ、電圧源、電流源の増分として、抵抗回路網の電圧源電流源として与え、先に求めてあるトランスミッタンスとイミッタンスを用いた重ねあわせによって、その電圧／電流の増分の説明が行える。

また、（１）にE_{Cn}，（２）にI_{Lm}を乗じ、全てを

加えあわせると、左辺は、キャパシタとインダクタに対するエネルギの出入りを表わし、右辺は、抵抗でのエネルギ消費と外部電源から（へ）のエネルギの供給を表わす。これを援用して、瞬時瞬時のエネルギの流れが説明できる。

電気機械系についても、同様の方法によって、とくにエネルギの流れに着目したグラフ表現と説明生成機構を試みている。

5．電磁気学の例

電磁気学に於いては、問題の表現に3次元ベクトル場や曲面、曲線の表現が欠かせない。ベクトル場を自由に自分で調べてまわることができるように、ユーザが、プローブ面の大きさと位置と方向を任意に設定すると、電磁気学の認識型と知識を利用して、この面の上に、面を貫く力線の断片の形で場を表示する。導体配置や電流面などとの関係でプローブ面の位置や方向を認識させるには、ユーザが位置を決められる点／平行光源による照明を当てることによって導体面あるいは適当な参照面の上に生ずる面の影（に相当するもの）を表示する。

また、問題解決のための知識としては、電磁気学の諸知識に加えて、場における面積分や線積分が特別に簡単になる状況をキャッチする知識が必要になる。

問題解決実行フェーズにおいて、多くの問題は、リアリティ条件等式知識の組合わせで解かれるから、等式関係を節点とし、変数を枝にもつ無向グラフを、新しい知識がみつかるたびに接続して表示してゆけば、3．、4．と同様な解決方針の支援ができる。

6．UNIXコマンドの例

ディレクトリをフォルダー、ファイルを書類ファイルとする通常のメタファを用いる。

UNIXのひとつひとつのコマンドは、オプションも含めると、ユーザの概念から見た場合、相当に多様な、しかも複合した仕事をすることができる。従って、コマンドの単位よりもっとプリミティブな固定された仕事をするエージェントいうメタファ[3]を用い、コマンドの行う仕事は、実はそうしたエージェントがつぎつぎに呼出されて仕事をするものとした方が分りよい。

具体的には、UNIX（のシェルならびにカーネル）の中には、フォルダ、ファイルを始めとする様々な対象を扱う様々なエージェントないしデモンが住んでいるとする。ユーザは、コマンドを通してエージェントに仕事をさせることはできるが、直接これらのエージェントに対して、仕事を命ずることはできない。

ユーザが入力した文字列を解読するエージェントは、それがコマンドであると判断すると、ディスパッチエージェントに解読結果を送り、しかるべきエージェントに指示を送らせる。リダイレクションがパースされると、ディスパッチエージェントが、標準入／出力に替るしかるべきファイルへ／からの入／出力を扱うエージェントに指示を送る。

このほかのエージェントの例としては、入力と出力を指定されたパイプを設定し、フィルタエージェントグループが流れ作業をするための準備をするもの、ファイルやフォルダを作りだすもの、その内容や構造を変えたり、内容をとりだしたり、それらに関する情報をとりだしたり、データをファイルに送り届けたり、システムの状態を変更するものなどがある。また、ファイルやフォルダには、その利用権状態に合わないアクセスを試みるエージェントが来るとこれを追い返すデモンが必要である。

エージェントとしてはシステムコールそのものを考えることも出来るが、あくまで、ユーザの理解（あるいは不理解）を表現するための比喩的なものと考える。

エージェントの仕事の結果、システムの状態、ファイルシステムの構造やファイルの内容、属性等が変り、こうした状況の一部を条件としてつぎのエージェントの仕事を可能にする。こうした時々の状況をコンテキストという[8]。

UNIX学習支援システムには、つぎのような認識型と、知識クラス、問題型を持たせる。

認識型として、まずエージェントの仕事の対象として、図1．の様なクラス階層がある（インスタンスを考える時でも、あくまで、ユーザの認識であって、実体といったものではないことに注意）

フォルダクラスのインスタンスとしてのフォルダでは、フォルダ構造という属性の値としてのフォルダとファイルの構造があり、ファイルクラスインスタンスにはデータ属性の値としてのデータがある。

ファイルシステムクラスは、フォルダとファイルの両クラスを統合するクラスで、名前、タイムスタンプ、

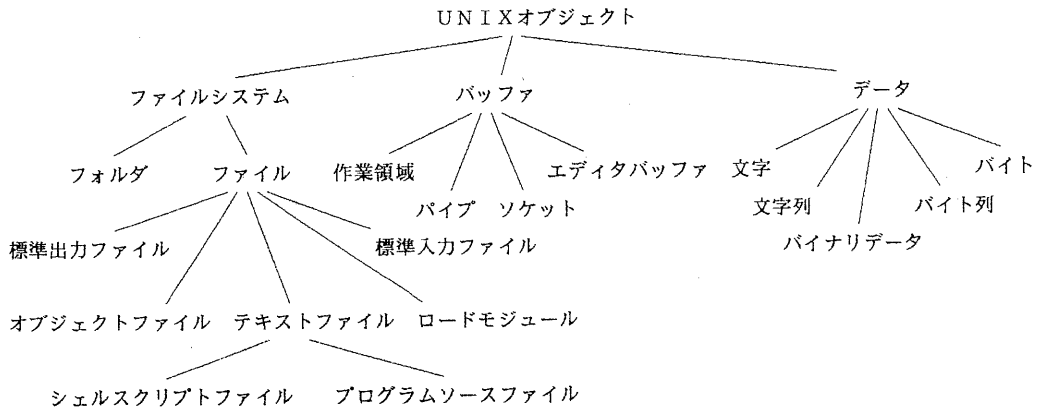


図１．ＵＮＩＸオブジェクト

アクセス属性、サイズ、所有者、グループ名など、フォルダ／ファイルに関して実際にはｉノードに書かれる情報を属性としてもつ。

次に、ＵＮＩＸオブジェクトあるいはシステムが、どのような状態にあるか－状況と言おう－を表わす認識型がある。例えば、

「データ D はデータ D1 とデータ D2 の結合である」、「ファイル F の内容にデータ D がある」、「作業領域にデータ D がある」、「入力ファイルが F1 で出力ファイルが F2 であるパイプ P がある」

などであって、述語表現として表わされ、コマンドシステムが達成すべきゴールやコンテキストの表現に用いられる。

また、知識としてはつぎのような型がある

（０）オブジェクトに関する知識、例えば、文字列と文字列を結合すると文字列になる、など。
（１）どのエージェントないしデモンがどのような仕事を受け持つかに関して、仕事が出来るためにコンテキストに要求する条件、結果としてのコンテキスト変更に関する知識、
（２）個々の（オプションを含めた）コマンド記述に対し、ディスパッチエージェントが、どのエージェントにどの順序で指示をだしてコンテキストの変化がどうなるかまでパースするための手順に関する知識
　－（０）（１）も利用する。
（３）コマンドスクリプトの解析のための知識

問題型として
（ａ）コマンドあるいはコマンドスクリプトを、コンテキストに関する前提条件と、結果としてのコンテキストの変更にまで解析する問題
　＋その解決を支援する方略知識

（ｂ）達成すべきゴールの記述－与えられる状況と達成すべき状況－から出発して、それを達成するコマンドあるいは、コマンドスクリプトを生成する問題＋その解決を支援する方略知識（エージェント、コマンド、スクリプトの３層の階層をなす操作が名乗りを上げる blackboard モデル）

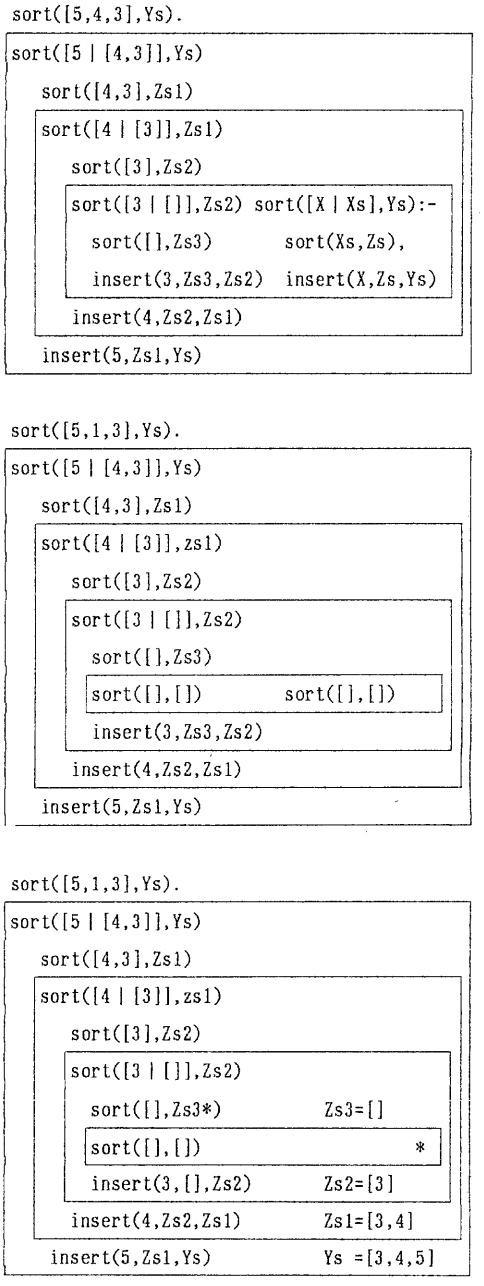
（ａ）の問題の支援の例：
　例えば、標準出力ファイルが、ＣＲＴディスプレイのとき、cat f1 f2 を解読させる。（０）の知識を援用しつつ（２）（１）の知識を検索させる。
　結局、これは次のようなエージェントアクションとコンテキストの進行に展開される。その途路で、CRT ディスプレイファイル crt、データ x 、ファイル f1, f2およびそれらの内容としてのデータ x1, x2 という認識インスタンスがつくられる。

前提：［crtは、ＣＲＴディスプレイ、crtは、標準出力ファイル、f1はファイル、f2はファイル、ファイルf1の内容はデータ x1、ファイルf2の内容はデータ x2、x1は文字列、x2は文字列］、
エージェント：ファイルの内容を読む（f1,x1）
［作業領域にデータ x1 がある］
エージェント：ファイルの内容を読む（f2,x2）
［作業領域にデータ ［x1 x2］ がある］
エージェント：データを結合する（x1,x2,x）
［作業領域にデータ x がある］
［データ x は データ x1 と データ x2 の結合、xは文字列］
エージェント：データを標準出力に届ける（x,crt）
［ファイル CRT にデータ x がある、CRT上でデータ xは読み取れる］
ここで、（０）（１）の型の知識と、CRTに住むデモンが役目を果たしていることが知られる。

（ｂ）の問題の例は省くが、はるかに高度な支援が要る。いずれの問題にしても、コンテキストの推移部分を強調した状況推移としてグラフに表わすことが必要であり、とくに、（ｂ）の支援では、エージェントやデモンが名乗りをあげるシーンをメタファで示すことが重要である。

７．Ｐｒｏｌｏｇプログラミング

Prologのプログラミングについては、以下の３つの支援を試作している。
（ａ）インタプリタの働きを原理的に理解することを助けるようなトレーサのメディアオブジェクトをつくっている。
例えば、sort([X | Xs],Ys):-
　　sort(Xs,Zs),
　　insert(X,Zs,Ys).
sort([],[]).
で、insert()は、すでに了解したので、結果だけをみることにして、？－sort([5,4,3],Ys).のトレースの途中から最後までを、図２．に示す。
（ｂ）プログラム構造理解支援のために、プログラムの構造に関する認識型と知識ベースを持たせ、具体的なプログラムのトレースを、構造記述によるトレースと重ねあわせて見ることができるように考えている。



＊ 部分が blinkingし、以下、マウスクリックするごとに、外の枠へ戻ってゆき、変数の値が決まる度にその位置が、また枠中のゴールが解決する度にその背景が、blinkingする。

図２．Prolog トレーサ

先の例について、再帰節の構造表現を示す。

後ろ向き再帰：

ゴール（Xs,Ys,リストXsの要素を大きさの順に並べたリストYs）

→簡単化ゴール（Xs,Xs1,リストXsの尾部 Xs1）

ゴール（Xs1,Ys1,リストXs1の要素を大きさの順に並べたリストYs1）

最後の変換ゴール（[Xs,Ys1],Ys,リストXsの頭部を、順序つけされたリストYs1の適切な位置に挿入して得られるリストYs）

ここに、一般にゴールは

ゴール（<与えられたデータ>、<結果として得たいデータ>、<両データが満たすべき関係>）

というかたちで表現される。

（ｃ）プログラム設計問題の支援をするには、

（０）引数として可能なデータ構造のクラス階層

（１）ゴールを記述する抽象された関係としての関係プリミティブという認識型のクラス階層

（２）プログラムの構造をクラス分けした構造プリミティブという認識型のクラス階層

（３）（１）に置かれている関係プリミティブの各々によるゴールを構造プリミティブを使って、他の関係プリミティブによるゴールで表現した知識の集り

（４）データ構造と関係の手がかりがついたサンプルプログラムの集合、

を必要とする。

まず、課題ゴールとして、与えられるもの（GIVEN）のデータ構造と得たいもの(DESIRED)のデータ構造をメニューから選ばせる。また、それらの間に成立すべき関係を抽象化して、メニューにある関係の接続として表わさせる。こうした関係の中には、未知の関係を未知述語として含む関係も多い。

登録された関係には、それをゴールとして実現する構造が知識としてあるからそれを引用すれば、上位レベルのプログラミングは完成する。続いて、未知の関係の部分を、再び抽象化により、メニューにある関係で表わすというように再帰的に続けられる。

もし、関係の知識ベースが不完全で、はじめから、あるいは途中から、適当なものが見つからない場合には、データ構造が対応し、関係として近いプログラム例を検索させる。自分のプログラムゴールの関係をことばと引数を使って表現させ、プログラム例を参考にプログラム構造を選ばせて、そのなかで、サブゴールの関係の記述をさせるが、その場合、簡単化や、最後の変換、途中の更新などにおけるGIVEN と DESIRED の関係名として、登録されているものから選ぶ。なければ、そのような関係名を暫定登録して、その部分の設計を別に行わせる。

［ｎクイーン問題の例］

GIVEN:自然数 N 、DESIRED:自然数のリストQs

関係: NNの行列Aがある。N人のクイーンを互いに攻撃しないようにAの各列に置く時の行番号のリスト Qs。

登録されている抽象関係としてつぎのものがあったとしよう。ここで、R(Q,Qo)を、互いに攻撃しないという関係とすれば対応する。

［長さ N のデータの列 Qs の各要素Qは、C(Q)を満たし、他の全ての要素Qoとある相互的－つまり、引数の順番を入れ替えても変らない－関係 R(Q,Qo) をもつようなもの］。

実現の構造：

ゴール（［N,R,C］,Qs,長さ N のデータの列 Qs の各要素Qは、C(Q)を満たし、他の全ての要素Qoとある相互的関係 R(Q,Qo) をもつ）

→ 縮小（N,N1,N1 は N マイナス１）

ゴール（[N1,R,C],Qs1,長さ N1 のデータの列 Qs1の各要素Qは、C(Q)を満たし、他の全ての要素Qoとある相互的関係 R(Q,Qo) をもつ）

最後の変換（[Qs1,R,C],Qs,C(Q)を満たし、Qs1の全ての要素Qoと関係R(Q,Qo)をもつQをそれに前置した Qs）

最後の変換のゴールは、登録されていないかもしれないが、

C(Q)を満たし、Qs1のすべての要素Qoと関係R(Q,Qo)をもつQ、

リストQs1 に Q を前置したリストが Qs

という関係の複合として表わされる。

従ってその実現の構造は、

ゴール（ [Qs1,R,C],Qs,C(Q)を満たし、Qs1の全ての要素Qoと関係R(Q,Qo)をもつQをそれに前置した Qs）

→ゴール（[Qs1,R,C],Q,C(Q)を満たし、Qs1のすべての要素Qoと関係R(Q,Qo)をもつQ）

ゴール（[Qs1,Q],Qs,リストQs1 に Q を前置したリストが Qs）

第2ゴールはありふれたものですぐに構造が検索されよう。第1ゴールも登録されているであろうから、構造はつぎのように登録されているだろう。

ゴール（[Qs1,R,C],Q,C(Q)を満たし、Qs1のすべての要素Qoと関係R(Q,Qo)をもつQ）

→C(Q),

ゴール（Qs1,Qo、リスト Qs1 から頭部 Qo 抽出）

条件分岐 R(Q,Qo)

yes：縮小（Qs1,Qs2,尾部）

ゴール（[Qs2,R,C],Q,C(Q)を満たし、Qs2のすべての要素Qoと関係R(Q,Qo)をもつQ

no : fail

ゴール（[[],R,C],Q,C(Q)を満たし、Qs1のすべての要素Qoと関係R(Q,Qo)をもつQ）

→success

あとは、C(Q)とR(Q,Qo)を実現すればよい。

かなり抽象的な思考を要求されるから、データ構造とプログラム構造の図解、関係の表現に工夫が要る。

8．作曲における和声進行の支援

基礎的な和声進行の仕方に関する知識ほもつほか、ユーザが進行の仕方を登録することができる。作曲の場面においては、こうした知識の中から、それまでの進行に部分的にマッチする和声進行の候補を検索し、その中からユーザに選ばせるとき、音符あるいは鍵盤図上での視覚的な提示とともに耳に聞かせて選ばせることができるように構成している。

一方和声進行を登録する際にも、音符を入力するだけでなく、鍵盤上の演奏あるいは、楽音そのものからも可能であるべきである。

9．あとがき

本稿では、学習者の問題解決を支援する知識ベースから制御されるメディアインタフェースオブジェクトの汎用的構成と、その利用について提案した。この仕組を用いれば、システム、あるいはシステムの支援によって学習者が、問題の対象世界の認識や知識について、知覚・運動感覚に訴えるメディア表現を生成し、さらに、問題解決実行手順をメタファによって提示し、こうした表現に対して学習者が介入して結果をみることができる。続いて現在進行中のいくつかの試作事例について、その構想を述べたが、この仕組により、問題解決と知識獲得の諸場面における対話が容易になり、学習者による方略や知識の発見を支援できることが期待される。

高度な問題解決において要求される抽象的な思考をどのようなメタファ表現によって支援できるかは、今後の課題である。

参考文献

[1]Itoh,K.,Itami,M.," A Computer-Assisted Knowledge Exploration Environment for Learning by Problem Solving",in Lewis,R. Otsuki,S.(eds.), Advanced Research on Computers in Education, North- Holland (1991)

[2]伊藤、伊丹、宮本：学習支援環境ＣＡＦＥＫＳにおける問題解決支援機構の試作 電子情報通信学会論文誌 第 75-A 巻第 2 号 （掲載決定）

[3]Minsky,M.:心の社会、安西祐一郎訳、産業図書

[4]Hori,K.,Ohsuga,S.:Assisting the Articulation of the Nebulous Mental World,in Jaakola,H,, Kangassalo,H.,Ohsuga,S.(eds) Advamces in Information Modeling and Knowledeg Bases,IOS Press(1991)

[5]Polya,G.:数学の問題の発見的解き方、上、下、みすず書房（1964）

[6]Watanabe,S. et al,"Teaching Circuit Analysis: A Mixed Initiative Intelligent Tutoring System and its Evaluation",in Lewis,R. Otsuki,S.(eds.), Advanced Research on Computers in Education, North- Holland (1991)

[7]Stallman,R.M.,Sassman,G,J.:Forward Reasoning and Dependency-Directed Backtracking in a System for Computer-Aided Circuit Analysis, Artif.Intell.,vol.9(1977)

[8]Wilesky,R.,Arens,Y.,Chin,D.:Talking to UNIX in English:An Overview of UC, Communications of the ACM , vol.27, no.6, (1984)