

解説

力学の CAI システム—PQRS†



安西 祐一郎†† 中村 久 肇†††

1. 力学における問題の理解

「問題が分かれば解けたと同じことだ」とよく言われるように、問題解決においては問題の理解がきわめて重要である。特に初等力学の問題では、利用すべき物理学の知識はかなり整然としており、いったん問題が理解されてしまえばそれを解くのは容易であるのに、物理現象を力学の問題「として」理解することが難しいことが多い。たとえば図-1の問題を考えてみよう。すなわち、テーブルの上に糸を巻きつけた糸巻きが置いてあるとき、糸の端を矢印の方向に引っ張ると糸巻きは((a)左に転がる、(b)右に転がる、(c)動かないのうち)どうなるだろうか。ただし、糸巻きはすべらないものとする。この問題は、力学を知っているはずの人にとってもそれほど簡単ではない¹⁾。ただ、その難しさは、問題の中に適切な物理学的構造を見だして力学的な定式化を行うことにあり、それが得られてしまえば簡単に解けることが多い。ここでは、上に太文字で述べたことを問題の理解と呼ぶ²⁾。

円形の二つのわくの真中に、しんをとおして、これに線が巻きつけてあります。いま、下に示すように線を引きばるとするとどうなるでしょうか？
ただし、円板はころがることはあっても、けっしてすべることはありません。

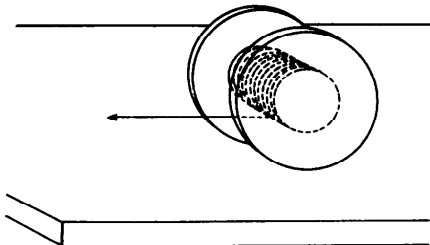


図-1 糸巻きの問題

† PQRS: A CAI System for Mechanics by Yuichiro ANZAI (Keio University) and Hisatoshi NAKAMURA (Fuji Xerox Corporation).

†† 慶應義塾大学
††† 富士ゼロックス

さて、初等力学の学習において最も重要なことの一つは、ニュートン力学の概念的知識が物理現象とどう関係するかを理解「しようとする態度」を養うことである。この態度なしでは学習された知識が机上の空論と化してしまうことは、受験勉強の弊害ではないかとしてたびたび語られているところである。

このような態度を養うことは本質的には生徒自身の問題であるが、それを助けるための一つの方法として、次の機能をもった CAI システムを構築することが考えられる：(1)力学の問題の物理学的表現を視覚的に呈示でき、しかも(2)その表現を因果的あるいは空間的な変換(すなわち力学の形式的世界でなく生徒の自然な認識の世界における表現の変換)によって別の表現に変換でき、さらに(3)それを繰り返すことによって、生徒が問題のいろいろな力学的構造を自分の認識世界における物体の変化と対応づけることができるようにする。ただし、上で、因果的変換とは、たとえば「箱Aが別の箱BにぶつかるとBはAと逆の方向に動く」といった、経験的な因果関係の知識に基づいて解釈される表現の変換をさし、空間的変換とは、空間的な位置関係の変化による表現の変換を意味する。

以下に概要を述べるニュートン力学の CAI システム PQRS (Physics Qualitative Reasoning System) は、上の(1)-(3)を念頭において、数年前から開発が進められたシステムである。PQRS では問題を生徒自身が作成して計算機に入力し、計算機が、運動方程式を立ててそれを解くとともに、その問題の力学的表現の因果的あるいは空間的な変換を繰り返すことによって問題を解く。このとき、問題の解決過程の運動方程式による表示と問題表現の視覚的呈示が一緒に、各段階ごとにディスプレイ上に行われる。問題を生徒が計算機に与えるというアイデアは、現在では Guidon-Watch, Sophie-III などによって採用されているが、われわれのシステムはそれらとは独立に考えられたものであり、また問題表現の変換規則は新しいものである³⁾。

2. PQRS の概要

2.1 PQRS における問題理解と解決の方法

ここでは、PQRS の概要について述べる。1. に述べたように、PQRS は問題構造の表現変換の手法を用いて力学の問題解決を行うシステムであり、表現変換規則として因果的および空間的表現変換の規則を用いることによって、CAI への応用における強力な説明機能を提供している。PQRS は、階層構造をもつプロダクション・システムを埋め込むようにしたオブジェクト指向型知識表現言語 OPHELIA (Object-oriented Production-embedded Human Everyday Language and Intelligence Accomodator)⁹⁾ を用いて書かれており、問題解決のトップ・レベルにおけるプランニングとその下で行われる問題構造の表現変換のための規則を、OPHELIA の階層的ルール・ベース機構によって実現している。さきにも述べたように、PQRS は、ブロック、滑車、斜面などを含む問題、さらに摩擦を含む問題に対して、運動方程式を立てて解く定量的方法と、問題表現の逐次的変換によって解く定性的方法による解決過程を並行的に提示する。このような説明機構によって、PQRS を利用する生徒が問題の力学的構造と方程式による解決過程を対比し、それをとおして問題の理解を促進できるようにしている。

2.2 問題表現の変換方略

PQRS では、ブロック、斜面、滑車などの物理学的対象を OPHELIA におけるクラス・オブジェクト、問題に含まれた対象をそれらのインスタンスとして表現する。また、力のような物理学的概念や、「上」、「下」、「左」、「右」といった位置関係の概念もクラス・オブジェクトとして表し、力の伝達や力の方向変換などをオブジェクト間のメッセージ・パッシングで表現する。

問題を解くためには、対象間関係や外力の記述から対象全体の挙動を推論する必要がある。PQRS における表現変換法では、いくつかの対象をまとめて一つの仮想的対象（「系」とよぶ）とみなし、問題表現を単純化しつつ得られた新しい表現に対して挙動の推論を行う。たとえば、外力が作用しているブロックがさらに隣接した別のブロックを押している場合、これら二つのブロックを一つの仮想的な質点と考えることができる。そこで、二つのブロックを新たな一つの系に置き換えることによって、ブロック間の抗力を考慮する必

要がなくなる。このことは、運動方程式の側では、考慮すべき方程式の数を減らすことに対応する。

系を生成するための表現変換方略としては、次の 4 種が使われている：

(1) **gravity-kei** 互いに糸で結ばれたいくつかの物理学的対象に作用している外力が重力だけであれば、それらの対象を一つの系にまとめる。

(2) **force-kei** 互いに接したいくつかの物理学的対象が一体となって運動する場合、それらの対象を一つの系にまとめる。

(3) **string-kei** 互いに逆向きの外力が作用している糸で結ばれた物理学的対象群を一つの系にまとめる。

(4) **pull-kei** 外力の作用した物理学的対象に糸で引っ張られる物理学的対象群を一つの系にまとめる。

これらの方略は、クラス・オブジェクト strategy のもつルールとして表現される。なお、上の 4 種のほか、摩擦力により一体となって運動する物理学的対象群を一つの系にまとめる手続きや、仮想的に系を生成する手続きがある。後者によって系が仮定された場合、PQRS は、問題解決過程で張力や抗力の存在をチェックし、系を構成する対象群が実は一体となって運動しないことが分かったらもとの別々の対象に戻して再計算を行う。

なお、表現変換による問題解決のある時点における「状態」は、物理学的対象と対象間関係の集合として定義される。したがって、表現変換方略は、状態の一部を系に書き換えることによって状態遷移を行う問題解決オペレータとみなすことができる。また、系の生成に使われた対象は遷移後の状態には陽に含まれなくなってしまうから、特定の対象に対して問題の答を求める（たとえばあるブロックの加速度を求める）ためには、その対象が他のどの対象とともにどの系に変換されたかを記憶しておくことが必要である。このため、PQRS は常時状態遷移の管理を行っている。

2.3 物理学的対象の表現

ここでは、物理学的対象の表現方法について少し具体的に述べる。次のような問題を考えてみよう：

「図-2 のように、重さがそれぞれ 20 kg, 10 kg の 2 個のブロック b1, b2 があり、滑車を通した糸 s1 でつながっている。b1 に左向きに 120 N の外力が加わっているとき、b2 の加速度を求めよ。」

このとき、ブロック b2 は PQRS の中ではクラス

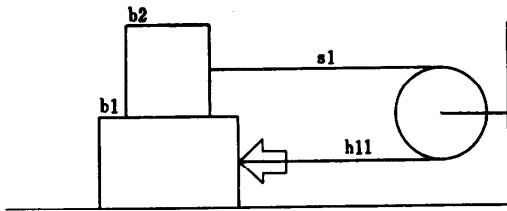


図-2 2個のブロックと滑車の問題

```

***** INSTANCE NAME: b2
          CLASS: block
          VARIABLES: (mass 10.0)
                    (contact (b1 s1))
                    (down (b1))
                    (right (s1))
  
```

図-3 ブロックのインスタンス表現例 (必要な箇所のみ)

block のインスタンスとして図-3 のように表現され、質量 (mass) や他の対象 b1, s1 との空間的位置関係が記述されている。ただし、図-3 は一部簡略化して示してあり、実際には up, right といった空間的位置関係は状態表現とのリンク (OPHELIA における推論制御の一方法) をとおして制御されている。

上の問題には外力が含まれているから、PQRS は「力」クラス force のインスタンスを生成し、それを「ブロック」クラス block のインスタンス変数 external-class に記述する。図-2 の場合は b1 にかかる左向きの力と、b1, b2 にかかる重力の三つの外力が存在するが、このうちブロックの動きに影響するのは b1 への左向きの力である。ブロックと滑車を支えている床と壁は動かないと仮定されているので、b1, b2 にかかる重力はその反作用とつりあい、キャンセルされる。また、b1 の左方向への運動は、糸と滑車により、b2 にとっては右方向の運動となる。このような運動方向の変換は、「滑車」クラス pulley と「糸」クラス string におけるメソッドとして表現されている。

2.4 問題解決の制御

ここでは、PQRS における問題解決の制御方法について、やや詳しく述べる。

PQRS における問題解決は、クラス・オブジェクト階層のトップ・レベルにあるクラス pqrs のもつルールによって制御される。たとえば、問題解決において現在ゴール・スタックの一番上にあるサブゴールが物理学的対象をまとめて系を作るというサブゴールであれば、表現変換のための推論がクラス strategy のもつルール群により行われる。このとき、strategy の上位クラスである pqrs のもつルール群も、OPHELIA

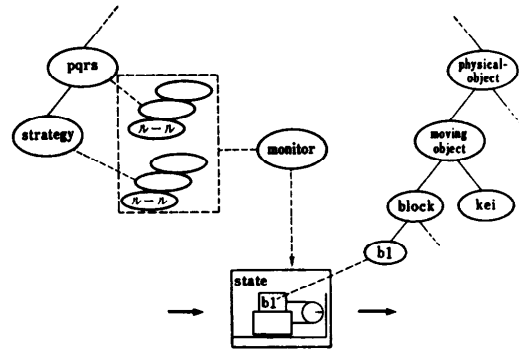


図-4 PQRS における問題解決の制御

のプロダクション・システム継承機構によって一緒に動く。この機構により、系生成のための手続き的知識を問題解決のための制御機構のもとで使うことが容易になっている。

また、さきに述べたように、PQRS では問題解決の「状態」を物理学的対象と対象間関係を表現するインスタンスの集合で定義する。このような「状態」は、クラス state のインスタンスとして表され、新しい系が生成されるたびに、state の新しいインスタンスが作られる。

一方、対象間の位置関係のような状態ごとに異なる情報は、OPHELIA のリンク機能を用いて対象と状態の間のリンクによって制御している。

また、状態遷移の管理はクラス monitor が行う。とくに、ルール記述が状態に依存しないようにするために、対象の空間的位置関係については、その情報にアクセスする必要が生じた時点で、monitor のメソッドによって現在の状態とリンクされている対象間の関係を求めている。図-4 に PQRS における問題解決の制御構造を示しておく。

3. 問題作成エディタとグラフィック機能

PQRS では、生徒自身がシステムの内部表現を意識せずに問題を作成できるように、問題作成エディタを用意している。問題作成エディタとしては、日本語処理システム (OPHELIA で書かれた HELPER システム) を利用したものなどが試作されたが、結局メニュー形式のものに落ち着いた。自然言語入力は生徒にとってもシステムにとっても現在のところ負担が大きく、また入力される問題を全く自由にするシステムには解けない問題が多くなってしまいう可能性があるのでその主な理由である。

(a)

```

>>>> Please input object (1 : table 2 : slope 3 : ceiling) ? 1
>>>> Please input object's name ? t1
>>>> Please input object (1:block 2:string 3:pulley 4:hand 5:slope) -> 1
>>>> Please input object's name ? b1
>>>> Please input weight (10.0 20.0 30.0 ...) (kg) ? 5
                                     —[A]
>>>> Please input object (1 : block 2 : string 3 : pulley 4 : hand 5 : slope) -> 1
>>>> Please input object's name ? b2
>>>> Please input weight (10.0 20.0 30.0 ...) (kg) ? 30.0
>>>> Please input support object (1 : b1 2 : t1) ? 2
>>>> Please input left contact object (1 : no-touch 2 : b1) ? 2
                                     —[B]
>>>> Please input object (1 : block 2 : string 3 : pulley 4 : hand 5 : slope) -> 2
>>>> Please input object's name ? s1
>>>> Please input contact end object(s) (1 : b1 2 : b2) ? 2
>>>> Please input direction from b2 to s1 (1 : right 2 : left) ? 1
                                     —[C]
>>>> Please input object (1 : block 2 : string 3 : pulley 4 : hand 5 : slope) -> 3
>>>> Please input object's name ? p1
>>>> Which direction from t1 to p1 (1 : right+up 2 : left+up) ? 1
>>>> Please input contact string (1 : no-touch 2 : s1) ? 2
                                     —[D]
>>>> Please input object (1 : block 2 : string 3 : pulley 4 : hand 5 : slope) -> 1
>>>> Please input object's name ? b3
>>>> Please input weight (10.0 20.0 30.0 ...) (kg) ? 10
>>>> Please input support object (1 : no-touch 2 : b1 3 : b2 4 : t1) ? 1
>>>> Please input contact string (1 : s1) ? 1
                                     —[E]
>>>> Please input object (1 : block 2 : string 3 : pulley 4 : hand 5 : slope) -> 2
>>>> Please input object's name ? a2
>>>> Please input contact end object(s) (1 : b1 2 : b2 3 : b3) ? 3
                                     —[D]
>>>> Please input object (1 : block 2 : string 3 : pulley 4 : hand 5 : slope) -> 1
>>>> Please input object's name ? b4
>>>> Please input weight (10.0 20.0 30.0 ...) (kg) ? 15
>>>> Please input support object (1 : no-touch 2 : b1 3 : b2 4 : t1) ? 1
>>>> Please input contact string (1 : a2) ? 1
                                     —[D]
>>>> Please input object (1 : block 2 : string 3 : pulley 4 : hand 5 : slope) -> 1
>>>> Please input object's name ? b5
>>>> Please input weight (10.0 20.0 30.0 ...) (kg) ? 5
>>>> Please input support object (1 : b1 2 : b2 3 : t1) ? 3
>>>> Please input left contact object (1 : no-touch 2 : b1 3 : b2) ? 3
                                     —[E]
>>>> Please input object (1 : block 2 : string 3 : pulley 4 : hand 5 : slope) -> 4
>>>> Please input object's name ? h1
>>>> Please input contact block (1 : b1 2 : b2 3 : b3 4 : b4 5 : b5) ? 5
>>>> Please push direction (1 : right 2 : left) ? 2
>>>> Please input magnitude (10.0 20.0 30.0 ...) (N) ? 200
                                     —[E]
>>>> Please input object (1 : block 2 : string 3 : pulley 4 : hand 5 : slope) -> 0
    
```

(b)

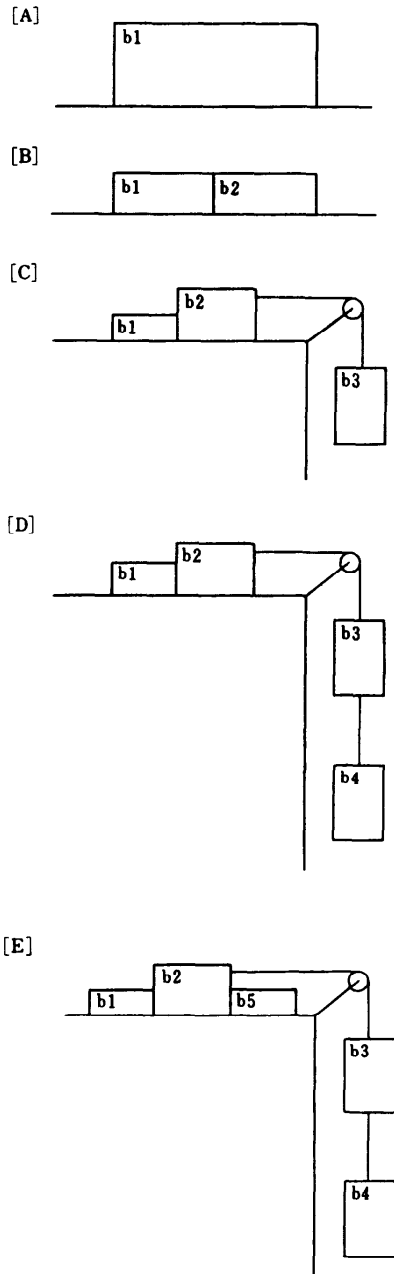


図-5 問題作成エディタの使用例 (a) メニュー入力, (b) 視覚的呈示 ((a), (b) の [A]~[E] がそれぞれ対応している.)

生徒は、システムが提示するメニューから入力したい対象やその位置などを選択すればよい。ただし、メニューによって問題の構成が限定されすぎてしまわないように、提示のしかたには相当の自由度をもたせている。システムは、ニュートン力学や空間関係に関する知識ベースに基づいて、生徒の入力から物理的対象や対象間関係のインスタンスを生成し、初期状態を構成する。

さらに、システム (Unix Berkeley 4.2 BSD 上で稼働) はパソコンに接続され、パソコン上の BASIC で書かれたグラフィック機能を用いて、図や式、日本語が表示できるようになっている。生徒が問題をメニューをとって入力していくにつれて、このグラフィック機能によりそれまで入力された問題部分が視覚的に呈示される。また、入力が終了した後の問題解決においても、表現変換による状態遷移のたびに、問題表現とその日本語による説明および式が、グラフィック機能を用いて呈示される。メニュー入力とその視覚的呈示の例を図-5(a), (b)に示す。

4. 処理例

例として、図-2 にあげた問題に対する PQRS の処理を説明しよう。PQRS は、まずブロックに作用する外力を求め、それをクラス force のインスタンスとして表現する。このとき、つりあっているとしてキャンセルできる力はキャンセルする (図-6(a))。

次に、さきに述べた表現変換方略の適用を試みる。

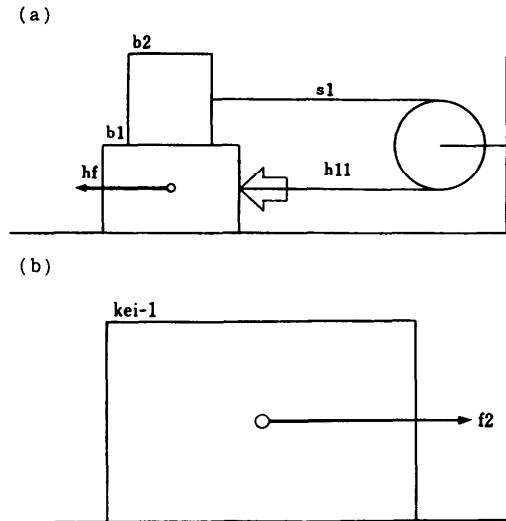


図-6 図-2の問題の PQRS による解決過程

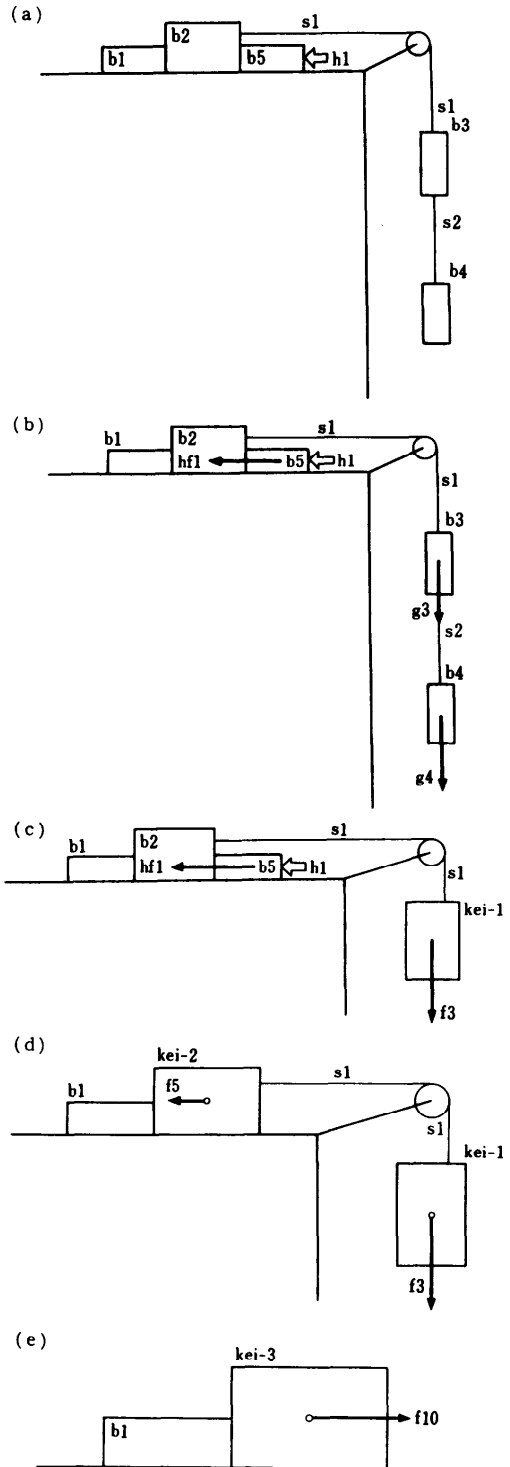


図-7 5個のブロックと滑車の問題の PQRS による解決過程

図-6(a)の場合は pull-kei が適用でき、b1 と b2 をまとめた新しい系として kei-1 が生成され、定性的には、1個のブロックに一つの右方向の力が作用する問題に帰着される(図-6(b)). 図-6(b)に示した kei-1 は仮想的対象であるから、そのもとになった b1, b2 における左右の方向性は意味をなさないが、求める答が b2 についてのものであることから、b2 を基準として、kei-1 に作用する力の方向を右としている。

この後 PQRS は、b2 が kei-1 に含まれることから、kei-1 の質量 $20\text{ kg} + 10\text{ kg}$ と kei-1 に作用する力 120 N から運動方程式

$$(20.0 + 10.0) * \text{variable-1} = 120$$

を立てて解き、b2 の加速度が右向きに 4 (m/ss) であることを呈示する。

もう一つ、もっと多くの物理学的対象を含む問題の例として、図-7(a)の問題における表現変換を考えてみよう。なお、この問題は問題作成エディタの例として示した問題(図-5(b)の[E])と同じものである。この例では、まず図-7(b)のように力のベクトルが生成される(ただし図-7(b)では力ベクトルの一部が表示されている)。次に表現変換方略 gravity-kei, force-kei, string-kei が順次適用され(図-7(c)~(e)), 系として kei-1, kei-2, kei-3 が生成されて、最終的には図-7(e)のような表現が得られる。

ここまでの段階は、ブロックの質量や力の大きさに依存しない。ただし、kei-3 にかかる力の方向は、kei-2 に作用する力(つまり、ブロック b2 にかかる左向きの力)と kei-1 に作用する力(ブロック b3 と b4 の重力から合成される力)との大きさを比較して決められる。kei-3 にかかる力の方向が図-7(d)のように(仮想的な)右であるとすると、ブロック b2-b5 の加速度は kei-3 に関する運動方程式を解くことによって求められる。また、ブロック b1 の加速度は、作用する力がつりあいの力だけであることから 0 であることが推論される。さらに、糸 s1 にかかる張力を求める場合には、遷移した状態をバックトラックし、図-7(c)の状態では kei-1, あるいは kei-2 に対する方程式を立てて解くことで求める。

5. 問題理解のための CAI と今後の課題

特に物理学においては、CAI システム上で得られた知識が生徒自身の自然認識とかけはなれてしまい、運動方程式の解き方はうまくなくても問題の構造を見抜くことができないことがあるとすれば、物理学を学

ぶ意味が十分でなくなる。CAI の中心的目的が十分な量の知識の生徒への伝達にあることは否定できないが、生徒の自然認識を主体とした物理世界の見方を養うという設計の理念もまた無視できないことのように思われる。上に述べたように、PQRS の主な目的は、生徒が自分で作った問題を、問題の表現変換によって解きながら、解決過程を表示、説明することによって、問題構造の理解を助けることであり、上に述べた設計の理念を実現するための1ステップだと考えられる。

ただ、CAI システムとしての PQRS には、まだいくつかの問題点がある。その第一は入力用インタフェースである。PQRS はメニュー・インタフェースを備えているが、システムの目的からいって計算機をまったく意識せずに問題を入力できることが望ましい。計算機を意識する部分が大きくなるだけ物理学自体について考える部分が少なくなるからである。そのためには、アイコンで物理的対象を視覚的に入力できるようなシステムがあるとよいであろうが、われわれはそこまで実装していない。

第二に、コースウェアの問題がある。生徒がただ適当に問題を作成して入力するのでは、学習効果が十分あがらないだけでなく、「システムに何ができないか試してやろう」といった物理学とは関係のない遊びになってしまう恐れがある。したがって、生徒への学習のガイドラインに沿って、生徒が自発的にシステムを使うようになることが望ましい。

第三の問題点は、第二の点に関係することだが、計算機から生徒への働きかけの問題である。PQRS の説明機構では生徒への個別の指導が行われていないが、十分な学習効果をあげるためにはある程度の相互主導型(mixed-initiative)システムにすることが望ましい。

第四に、現在 PQRS はニュートン力学のごく狭い問題領域しか扱わないが、知識の領域を拡張する必要がある。われわれは算数の応用問題に対しても同様の理念に基づく CAI システムを試作しているが⁶⁾、力学に限っても、特に浮力、振子、剛体など、物理学の初心者にとって理解しにくいのが、今のところ PQRS で扱っていない問題はたくさんある。これらを PQRS の枠組みに入れることは、今後の重要な課題であると考えられる。

参 考 文 献

- 1) Anzai, Y. and Yokoyama, T.: Internal Models in Physics Problem Solving, *Cognition and Instruction*, Vol. 1, No. 4, pp. 397-450 (1984).
- 2) 安西祐一郎: 問題解決のための知識の獲得—表象構成のための知識をめぐって—, *システムと制御*, 29 巻 1 号, pp. 8-15 (1985).
- 3) Anzai, Y.: Architecture for Problem Understanding in Physics Problem Solving, Paper presented as an invited talk at the 3rd International Conference on Artificial Intelligence and Education, LRDC, University of Pittsburgh (1987).
- 4) 安西祐一郎, 近藤公久: 階層構造を持つルール・ベース型表現を埋め込んだオブジェクト指向型知識表現言語 OPHELIA とその応用, *コンピュータソフトウェア*, Vol. 3, No. 3, pp. 43-60 (1986).
- 5) 安西祐一郎: 人間の認知構造を考慮したインタラクティブ問題解決システム, 福村晃夫編, *知識情報処理ハンドブック*, 14 章, オーム社, 印刷中.
(昭和 63 年 8 月 11 日受付)