知的ＣＡＩシステム

大阪大学　産業科学研究所
溝口理一郎

１．まえがき

コンピュータを導入して教育を支援するＣＡＩシステムが注目を集めている．ＣＡＩは知識を伝達する手段としての教育を支援するものと、学習者の自発的な学習意欲を促進するための環境を与えるものとの２つに大別することができるが、ここでは人工知能技術の導入が盛んに行われている前者について述べる．

ＣＡＩシステムの開発は，高性能マイクロコンピュータの発達に伴って盛んに行われているが，市販されている多くのシステムはDrill&Practice型であり，学習者の理解の状態に関係なく問題が提出され，最後に解答の評価が出力される程度のものである．既存のＣＡＩシステムが柔軟さに欠けることは従来から指摘されており，その最大の原因は学習者モデルの欠如であると考えられている．ＣＡＩは理想的には教師という教育の専門家が行う行為を模擬するものであり、１つのエキスパートシステムとみなすことができる．この考えに基づき知識処理技法を導入したＣＡＩは特に知的ＣＡＩと呼ばれている．既存のＣＡＩシステムが画一的であり、単に問題集がコンピュータの中に入っただけで、「電子頁めくり機」と悪口を言われたのに対して、知的ＣＡＩは学習者の理解の状態に適応する柔軟な個別指導の実現を目的としている．

本稿では，知的ＣＡＩの考え方を紹介すると共に，システム構成の中で特に重要な役割を果たしている学習者モデルに関して述べる[5][6].

２．知的ＣＡＩシステムの構成と概観

知的ＣＡＩは一般に図１に示すフレームワークを持っている．学習者モデルモ

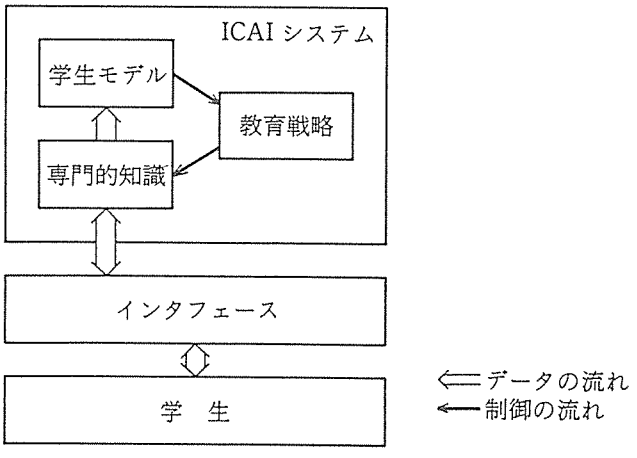


図１　知的 CAI システムのフレームワーク

ジュールは、学習者に与えた問題とそれに対する回答から教材に関する学習者の理解の状態を表すモデルを作成するものであり、知的ＣＡＩシステムの心臓部と言える．専門知識モジュールは教材モジュールとも呼ばれるが、教授すべき知識（教材）を持ち、問題を解くことは勿論、学習者の解答の評価も行う．個人指導モジュールは３つのサブシステムの中で最も動的なモジュールであり、システム全体の制御を司る．特に、学習者モデルに基づいて学習者の理解の状態を把握して、誤解があればそれを正すために必要な指導方針を立て、適切な説明、或は問題を与える．インタフェースは学習者が用いる自然言語とシステムの内部表現との変換を行う．

知識処理技術はこれらの全てのモジュールにおいて用いられている．その代表的なものが知識表現技法であるが、知的ＣＡＩシステムのなかでの知識表現の主要な対象としては、学習者モデル、教材、教育戦略の３つがある．これらを表現する手段として、意味ネットワーク、述語論理、フレーム、プロダクションシステム、スクリプト等の様々な方法が適用されている．学習者モデルの表現には、教師が持つ知識の部分集合として表すオーバーレイモデル，教師の知識に摂動を与えたものとして表す，摂動法やバグモデル等が用いられてきたが，最近帰納推論を適用して，問題とそれに対する学習者の解答の対を説明する一般則として学習者モデルを形成する方法が提案されている．又、システムと学習者とのインタフェースとして自然言語処理技術が応用されている．

知的ＣＡＩシステムとしては，南米の地理を教えるＳＣＨＯＬＡＲ，加減算の方法を教えるＤＥＢＵＧＧＹ，電子回路の故障診断法を教えるＳＯＰＨＩＥ，ＭＹＣＩＮの持つ知識（感染症の診断法）を教えるＧＵＩＤＯＮ，ＬＩＳＰのプログラミングを教えるＬｉｓｐ　Ｔｕｔｏｒ等が代表的であるが，多くは研究室レベルであり，実用に耐えるシステムは少ない．

３．知的ＣＡＩのための汎用フレームワーク

知的ＣＡＩへのアプローチとして，特定の教材をもとにして具体的なシステムの開発を行うボトムアップな方法論と，教示兄依存しない汎用の枠組みと必要な基盤技術の開発を行いトップダウンなアプローチとがある．本節以降では，筆者らが行っているトップダウン方法論を例に採って，知的ＣＡＩの本質を明らかにしたい．

専門家の知的な問題解決を模擬するエキスパートシステムは，プロダクションルールの形で表現された専門知識を蓄える知識ベースと，それを解釈・実行する推論エンジンとを持っている．この推論エンジンはルール表現に適した経験的知識の効率的な利用を可能にしたが，さらに知識のドメインに依存しない汎用の性質を持つことから，ルールベースに基づく一つのエキスパートシステム構築方法論を提供している．

一方，既に述べたように，知的ＣＡＩは教育に関するエキスパートシステムと見なすことができるが，そのための推論エンジンに関する検討は十分とは言えない．推論エンジンと関連の深い概念に汎用フレームワークがある．推論エンジン

の設計は，対象とする問題の構造を反映した形で行われるべきであるが，知的ＣＡＩのための汎用の枠組みを考察することは，システムの本質的な構造を理解する上で重要であるだけでなく，その考察の過程において知的ＣＡＩのための推論エンジンの概念が明らかになると考えられる．

汎用フレームワークの設計は，対象領域で本質的なタスク（以下では汎化タスクと呼ぶ）を見極め，それを処理するために必要十分なモジュールをフレームワークの構成要素（以下ではビルディングブロックと呼ぶ）として組織化することによって行われる．以下では，知的ＣＡＩのための汎化タスクを整理し，それに対応するビルディングブロックの構築原理について述べた後，推論エンジンの具体例を示す[10][13]．

３．１　汎用フレームワーク[8]

知的ＣＡＩシステムの３つの基本モジュール，学習者モデルモジュール，個人指導モジュール，教材知識モジュールは教育におけるタスクを分類するうえで，妥当な大分類と考えられる．表１に，これら３つの各モジュールごとに汎化タスクのクラスを示す（全てを網羅的してはいない）．表１中の各項目は，汎化タスクのクラスを示しているもので，汎化タスクに一意に対応していない．例えば，(T2)教育戦略というクラスには，手続き的な教材を教育する場合に必要な汎化タスクと，宣言的な教材を教育する場合に必要な汎化タスクが含まれるべきである．

汎用フレームワークの最終的な目標は，適切な汎化タスクを設定し，それぞれをビルディングブロックとして実現し，それらの統合環境を提供することにある．

表１．知的ＣＡＩにおける汎化タスクのクラス

(S) 学習者モデルモジュール
(S1) 学習者モデルの解釈･実行
(S2) 学習者モデルの構築
(T) 個人指導モジュール
(T1) 誤り分析
(T2) 教育戦略
(T21) 宣言的知識の教育
(T22) 手続き的知識の教育
(T23) 双方向対話
(E) 教材知識モジュール
(E1) 教材知識の解釈･実行,
(E2) オーサリング支援,
(E3) 教材シミュレータ,
(E4) 問題の自動生成

各タスクが，適切な抽象レベルで処理されれば，個々の対象領域において生じる

タスクは，その抽象レベルにおいて共通のビルディングブロックで処理することが可能になり，汎用性の高い統合システムが構築できる．また，ユーザは，適切な抽象レベルの用語を用いて知識をコーディングすることが可能になる．

3．2　ビルディングブロックと推論エンジン

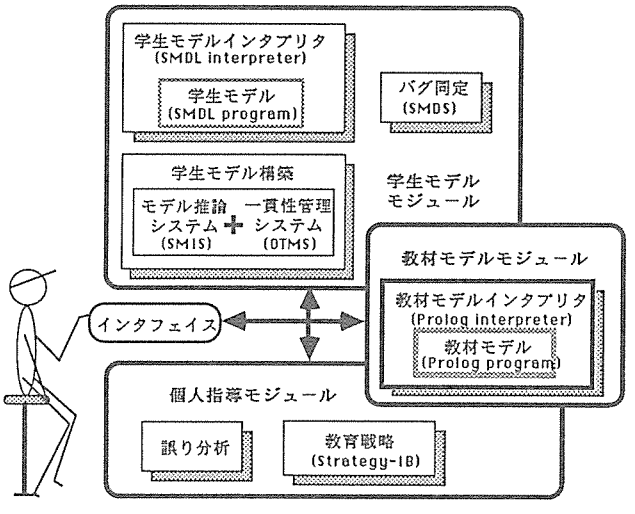
汎用性の高いビルディングブロックを開発するためには，対応する汎化タスクの一般的な問題空間での問題解決器を設計する必要がある．これにより，対象領域の問題空間の記述さえ与えられればビルディングブロックは汎用性の高い方法でその問題を解決する推論エンジンを提供することができる．さらに，この枠組みのうえに対象固有の知識を導入することによって，より知的な問題解決が可能になる．従って，ビルディングブロックの設計手順は次のようになる．

(1) 汎化タスクの問題空間の形式的定義
(2) (1)上での問題解決器（推論エンジン）の設計
(3) 対象領域に固有の知識の導入

例えば，Prologを表現言語として学習者モデルを構築する例をとれば，

(1) 空間として，述語で表現されるPrologのプログラム
(2) 推論エンジン（問題解決器）として，Prologプログラムの合成システム
(3) 教材に固有の知識の導入法として，学習者が誤り易い述語の優先的使用等

となる．このようにしてビルディングブロックを構築すれば，設定された空間の中での汎用の推論エンジンと教材に依存したモデル構成の効率化のための知識とが融合され，知的ＣＡＩのための強力な部品が実現される．図２に筆者らが開発中の汎用フレームワークを示す．以下では，学習者モデル表現用ビルディングブロックを例にとって，推論エンジンの具体例を述べる．



SMDL : *Student Model Description Language*
SMIS : *Student Model Inference System*
SMDS : *Student Model Diagnosis System*
OTMS : *Oracle"s Truth Maintenance System*
Strategy-IB : *Instance-Based Strategy*

図２　知的 CAI のための汎用フレームワーク

4．　学習者モデルと推論エンジン

4．1　学習者モデル

人間相互のインタラクションが相手の理解状態を把握することなしには成立しないことからもわかるように，教師が学習者の状態を正確に把握して初めて適切な教育が可能となる．この意味で，学習者モデルは知的ＣＡＩにおいて高度な個別教育を行うための本質的な役割を担っていると言うことができる．このような認識のもとに，これまで学習者モデルに関する多くの研究がなされてきた[5]．以下に学習者モデルを考える際に有効な視点について述べる．

4．1．1　学習者モデルの視点

(a)バグに対する考え方

学習者の理解状態をどこまで把握しようとするのかということはモデル構築における基本的な視点である．そもそもバグとは何か，そしてそれをコンピュータ上でどの様に表現するのかという問題を考えなければならない．知識の不足，誤った理解，誤った知識を持つに至った過程，などバグを追求する深さの問題，言い替えれば，バグを表現する際に，それを再現するだけでよいのか，あるいはバグが発生した理由の説明もする必要があるのか等ということはモデル構築の本質的な問題である．

(b)モデル表現

モデル構築の方法には，予めバグを整理しておき，学習者の誤りをそれらの組合せで表現する方法と，教材知識をバグの入り込む余地のない基本的なプリミティブに分解しておき，それらの組合せとして学習者の理解状態をそのまま表現する方法がある．前者は事前の準備が大変であるが，バグの概念を用いてトップダウン的にモデル化を行うので効率はよい．後者の方法にはバグの概念は存在せず，学習者の振舞いを素直にボトムアップ的な方法で表現しようとしている．

これに関連して問題となるのが表現プリミティブの大きさである．モデル表現の精密さと探索（モデル構築）の効率は相反する関係にあり，適切なトレードオフが必要である．

(c)実行可能性

学習者モデルの検証を行うにはモデルが出す回答と学習者の回答とを比較するのが最も妥当な方法である．従って，信頼性の高いモデルの構築にはモデルの実行可能性は不可欠である．さらに，学習者モデルを教育の過程で積極的に利用しようとするとき，学習者の振舞いをできるだけ正確に把握し，かつ予測することが望ましい．例えば次に出すべき問題を決定するには，実際に学習者モデルで問題を解いてみて期待されるように誤るような問題を選択する必要がある．従来のように，単元や項目の習熟度を数値で評価するモデルではこのような木目の細かい出題はできない．

(d)ノイズ

学習者モデルは学習者の振舞い，具体的には問題に対する回答，をデータとして構築されるが，その中には次に挙げる３つの原因によりノイズが混入する．

①表現言語が人間の理解や推論過程の近似に過ぎないこと.
②学習者が首尾一貫した振舞いをすることはなく，不注意による誤りなどが含まれること.
③学習が進むにつれて学習者の理解が変化すること．このため，ノイズを含むデータからモデルを構築することを考えなければならない.

4．1．2 学習者モデルの分類

(a)オーバーレイモデル[7]

楽観的にみれば，学習者は学習が進展すると共に知識が増加し，最終的には教師の知識と一致する．オーバーレイモデルはこの考えに基づき，学習者の知識を教師の部分集合として捉えるモデルである.

(b)差異モデル[3][15]

学習者と教師の解法の差異に注目して，学習者が未習得の知識を検出するという立場に立つモデルである．能力的にはオーバーレイモデルと同等である.

(c)バグのモデル

オーバーレイモデルや差異モデルではバグ（誤り）という概念が希薄であった．この型のモデルはバグをモデル化する事に重点が置かれている.

(c-1)数え上げ型

予め様々な誤答を解析し，誤った回答手順を数え上げてネットワークの形で表現するモデルである．バギーモデル[1]とも呼ばれる.

(c-2)再構成型

①バグの概念あり[4]

事前に学習者の多くの誤答を解析して誤った基本的手続きを整理し，正しい基本知識と共にモジュール構造で蓄積しておき，それらの組合せとして学習者の理解状態をモデル化する.

②バグの概念なし[10]

誤った知識に限定するのではなく，学習者が用いると考えられる最も詳細なレベルの基本手続きを整理し，それらの組合せとして学習者の理解状態を表現する．従って，システムは何が誤った知識かは知らない.

(c-3)原因分析型[2]

上に述べたモデルは，深くてもバグを再現するところまでで，バグが発生した原因にまでさかのぼってモデル化するものはなかった．この方法では学習過程の認知モデルに基づきバグ発生過程をもモデル化する.

筆者らが開発中のモデル構築手法は(c-2)②に分類される帰納推論に基づくものであるが，以下に推論エンジンという観点からその概要に付いて述べる.

4．2．帰納推論に基づく学習者モデル構築法

4．2．1 Prologによる学習者モデル表現

Prologを学習者モデル記述言語として採用すれば，学習者モデルはPrologのプログラムとなり，学習者の誤解はプログラムの虫となる．従って，教育はProlog

プログラムの合成とそのバグ取りの問題として定式化される．

酸と塩基（アルカリ）の化学反応式の教材を例にとり，Prologによる学習者モデル表現の例を示す．

```
酸(im, so(4), 2).
塩基(im, ca, 2).
沈澱(im, ca, so(4)).
塩(im, X, Y) :- 塩基(im, X, _), 酸(im, Y, _).
```

第一引数は理想モデル(ideal model) の知識であることを示している．これらは，SO_4が２価の酸であり，$CaSo_4$が沈澱し，酸と塩基の結合が塩であることを表している．ここで，学習者が塩という概念を酸と塩基の結合でかつ沈澱するものであると誤解していれば，

```
塩(sm, X, Y) :- 塩基(sm, X, _), 酸(sm, Y,_), 沈澱(sm, X, Y).
```

のようにモデル化される．ここで，第一引数のsmは学習者モデルの知識であることを示している．このように，Prologを用いることにより教材知識と学習者モデルを表現することができる．

４．２．２　Prologプログラム合成システム

（１）概略

学習者モデル構築タスクのための推論エンジンは，Prologを学習者モデル表現言語として用いれば，Prologプログラムの合成システムとなることは既に述べた．ここではPrologプログラムの合成アルゴリズムMIS（Model Inference System)[16] を採用した推論エンジンの例を示す．学習者モデルの構築は，学習者の振舞い（問題とそれに対する解答の集合）からそれを説明するモデルを構築するという点で，本質的に帰納推論の問題となるが，MIS は有限個の正の事実と負の事実を入力とし，それらを証明するPrologのプログラムを合成する帰納推論システムである．ここで，正の事実や負の事実とは

```
沈澱（ca, so(4)）, true
沈澱（pb, co(3)）, false
```

のようにPrologの事実とその真理値を組にしたもので，trueを持つものが正の事実，false を持つものが負の事実と呼ばれる．これらは，例えば「ｃａｓｏ$_4$は沈澱しますか？」と学習者に問うことによって得られる．プログラムが負の事実を証明してしまうとき，そのプログラムは強すぎると言われ，正の事実の証明に失敗すると弱すぎると言われる．MIS のアルゴリズムの概略を図３に示す．強すぎるプログラムを弱める時には誤りの原因となったホーン節を見つけるＰＤＳ(Program Diagnosis System)，弱すぎるプログラムを強めるときには，追加すべきホーン節を合成する精密化作用子がそれぞれに必要となる．いずれの場合にも，必要が生じれば事実の真理値に関する質問が行われるが，教育的にみて妥当なものであり，積極的に問い合わせることによって学習者モデルを構築していくという方法論を提供している．MIS の動きは基本的には演繹推論に基づく帰納推論であり，精密化作用子が候補節を生成する操作と，それらの候補節がそれまで入力さ

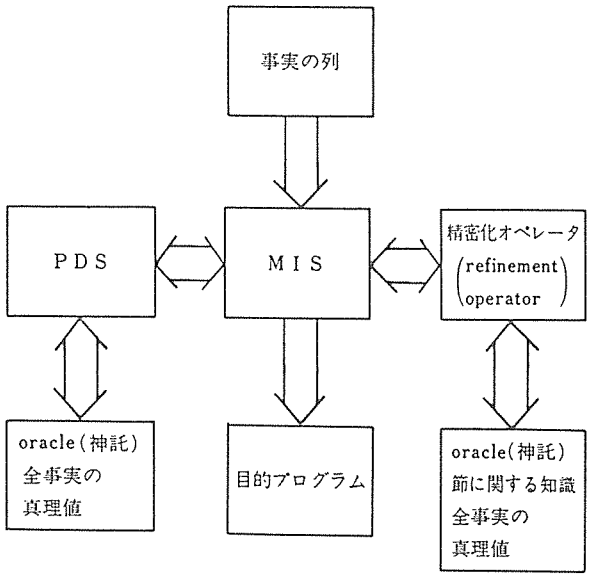


図3　MISのアルゴリズムの概略

れた事実を正しく証明できるかを検証する（演繹推論）操作が中心となっている．

（２）推論エンジンの強化

Prologは宣言的な性質を持ち，データとしての取扱も容易であると同時に，実行可能な手続き的性質も持っているため，学習者モデルの表現言語に適しているが，真と偽の２値しかとらないので，学習者が分からないという状態を表現することができず，必ずしも十分とは言えない．そこで筆者らは学習者モデルの表現力を強化するために，{true, unknown, false, fail}の４つの真理値を持つ言語SMDLとそれを用いた帰納推論法の開発を行っている[10][12]．true，unknown, falseはそれぞれ学習者がある事実を正しいと思っている，知らない，間違いと思っていることを示す．そしてfailはシステムが学習者の理解状態を知らない事を意味する．この拡張によりモデル表現が自然で強力になっている（これは，例えば学習者モデルの初期状態が空のとき，「システムは学習者が何も知らないと思っている」と考えるより，「システムは学習者の状態を知らない」と解釈する方が自然であることからもわかる）．SMDLはPrologを拡張したものになっており，モデル推論もMISを拡張したSMISとして実現されている．

さて，学習者の応答全てを説明するモデルの構築を行っている筆者らの方法では，ノイズ対策として誤ったデータからの帰納推論，即ち非単調な帰納推論が必要となる．非単調推論では，真理値が確定しない命題（仮定）と確定しているものとを区別して管理し，矛盾が発生した際に，その解消に貢献し得る仮定を効率良く求める機構が用意される．このようなことを行うために考えられている機構は一般にＴＭＳ（Truth Maitenance System)と呼ばれている．ＴＭＳに基づく学習者モデル構築システムでは，モデル構築の途中経過の情報をＴＭＳに与え，矛

盾（モデル構築の失敗）が起きればそれをＴＭＳに伝える．ＴＭＳは推論過程を解析することによって，矛盾の解消に必要な仮定を同定し，その真偽に関する信念を返す．より具体的に言えば，ＴＭＳは学習者モデル構築システムが矛盾解消のために訂正すべき学習者の応答の候補を教えるわけである．筆者らは，de KleerのATMS [14]を導入して学習者モデル構築方式の強化を行っている[10][11]．現在の主な学習者モデル構築技法は一つの問題と解答の組の情報に基づいているため，この問題の重要性は正しく理解されていないが，いずれ学習者モデル構築の基盤技術として認識されるであろう．

５．統合化アーキテクチャ[9]

これまでに述べた学習者モデル構築技法を組み込んだフレームワークに基づいて，現在，地理，１次方程式，化学反応の３つの教材に関する知的ＣＡＩを開発中であるが，この開発を通して明かとなった問題としてシステム全体のスケジューリング問題がある．従来筆者らは，スケジューリングは個人指導モジュールが行うという程度の認識しか持っていなかったが，開発を進めるに連れて実際に学習者に介入するという意思決定を行うことの困難さと重要性を認識するに至った．本節では，システム全体の動作を規定するための設計されたスケジューリング機構について述べる．

５．１　スケジューリング問題

教育を効果的なものにするためには既に述べたフレームワークの各ビルディングブロックを効率よく動作させなければならない．フレームワークの動作モデルを図４に示す．スケジューラは正確な情報を収集すると共に，これらの動作を効率よく制御する．ここでも教師が持つ知識は重要な働きをすることは言うまでもない．

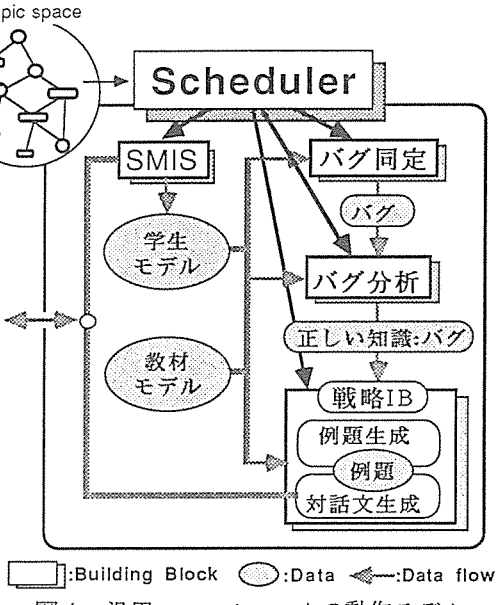


図４　汎用フレームワークの動作モデル

教材には知識間の依存関係などの複雑な構造が存在し，教材項目間の連続性や困難さの関係等と学習者の理解状態とを考慮しながら教育を進めなければならないし，学習者モデルの構築終了の判定，適切な介入のタイミングと教授の戦略等，考慮しなければならない多くの観点がある．スケジューリングシステムはこのように知的ＣＡＩシステムの行動全体を管理する，重要な役割を持っている．

5．2　スケジューリング機構

スケジューリング機構を設計するためには，スケジュールのために参照すべき知識を同定しなければならない．そのための知識は大きく２つに分類することができる．

(1)ビルディングブロックの能力に関する知識

各ビルディングブロックの仕様に関する知識であり，基本的には個々のビルディングブロックの機能とそれが適用可能な問題空間に関する知識である．

(2)教育学的知識

抽象度の高い知識であり，技術的観点とは独立にタスクやトピックの教育的妥当性を評価する知識である．

これらの知識の特性を考慮して，設計したスケジューリングのためのアーキテクチャを図５に示す．このアーキテクチャ上での動作の１サイクルは，情報の収

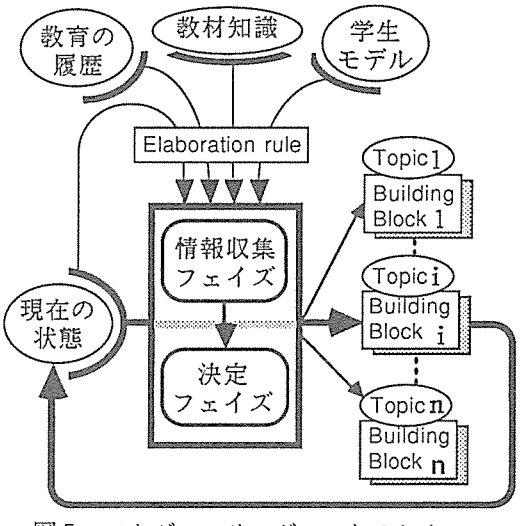


図５　スケジューリングアーキテクチャ

集フェイズと決定フェイズとの２つのフェイズからなる．情報収集フェイズではプロダクションルールの形式で記述された知識に従って，学習者モデル，過去の教育履歴，トピックの連続性などの情報を収集する．決定フェイズではそれらの情報を集約して，次に採るべき適切な行動を決定する．現在，この機構の洗練を進めると共に，知識の記述方式と整理を行っている．

6．むすび

知的ＣＡＩのためのフレームワークと学習者モデル構築のための推論エンジンに関して述べた．推論エンジンは単なるプロダクションシステムではなく，知的ＣＡＩシステムの本質的な構造を反映した汎用フレームワークの形で整理された，ビルディングブロックの一部として定めるべきものであることは重要である．知的ＣＡＩは大規模な知識システムであり，その実現には多大の努力が必要となる．システム構築の部品となり得る汎化タスクを適切に同定し，タスクの特性を反映した，効率のよい推論エンジンを設計し，システム全体を制御する枠組みを開発することは知的ＣＡＩシステム研究の発展に大きく貢献するものと思われる．

最後に，今後の課題をまとめてむすびとする．

（１）学習者モデルと認知モデル

学習者モデルをより優れたものにするための，人間の問題解決過程の認知科学的モデルの開発．

（２）教育戦略

柔軟な個別指導を行うためには，学習者モデルで表現された学習者の理解状態に適応して最適な指導方針を動的に生成し，次に出題すべき問題の決定とその生成、再教育すべき単元の決定などを行わなければならない．このための教育戦略の開発，及びその汎用性の検討は重要である．

（３）優れた対話インタフェイス

教師と学習者の接点である対話を円滑に進めるための双方方向性の対話制御方式の開発，及び学習者の理解の程度に適した説明文を生成する技術の開発．

（４）多様なシステム

学習意欲の促進に貢献する，環境型のシステムを含めた教育における多様な教育形態を支援するシステムの開発．

（５）実用性への配慮

ＣＡＩシステムの利用者の多くは子供であり、種々の制約や約束ごとに従ってくれることは期待し難い．そのため、知的ＣＡＩシステムの実用化には特にインタフェイスの頑強さ（robustness）が要求される．

（６）実証システムの開発

遠い将来の話だけではなく，現時点で役に立つ知的ＣＡＩを開発し，知的ＣＡＩの有効性を実証すること．

（７）構築支援ツールの開発

以上全ての課題を満足した知的ＣＡＩシステムの構築を支援するためのツール（オーサリングシステム）の開発は知的ＣＡＩ研究の最終課題とも言うべきものであろう．

＜参考文献＞

[1]Brown, J.S. et al.:Diagnostic models for procedural bugs in basic mathematical skills, Cognitive Science, Vol.2, No.2, pp.155-191,1978.

[2]Brown, J.S. et al.:Repair theory:a generative theory of bugs in procedural skills, Cognitive Science, Vol.4, No.4, pp.379-426, 1980.

[3]Burton, R.R. et al.:An investigation of computer coaching for informal learning activities,Int. J. of Man-machine Studies, Vol. 11, No. 11,pp.5-24, 1979.

[4]Burton, R.R.:Diagnosing bugs in a simple procedural skill, Sleeman, D.H. et al. eds., Intelligent Tutoring Systems, Academic Press, London, pp.157-183, 1982.

[5]溝口, 角所：知的ＣＡＩにおける学習者モデル, 情報処理, 29,11,pp.1275-1282, 1988.

[6]大槻：高度個別教育における知識情報処理－ＩＴＳの機能と振舞い－, 人工知能学会誌, Vol.1, No.2,pp.196-201, 1986.

[7]Carr, B. et al.:Overlays:A theory of modelingfor computer aided instruction, MIT AI Memo 406,1977.

[8]池田他：知的ＣＡＩのためのフレームワークの検討－学習者モデル, 帰納推論, 教育戦略－, 情報処理学会, 「教育におけるコンピュータ利用の新しい方法」シンポジウム, pp.49-58, 1987.

[9]池田他：知的ＣＡＩのためのビルディングブロックとその統合環境について, 情報処理学会「教育における知的方法」シンポジューム,63-70,1988.

[10]池田他：学生モデル記述言語SMDLと学生モデルの帰納推論アルゴリズムSMIS, 電子情報通信学会論文誌, D-II,Vol.J72-D-II, No. 1, pp.112-120, 1989.

[11]池田他：仮説型モデル推論システム, Proc. of the Logic Programming Conference '88,pp.47-56, 1988.

[12]Ikeda, M. et al.:Design of general frameworkfor ITS, Proc. of ITS'88,pp.82-89, 1988.

[13]河合他：論理プログラミングと帰納推論による汎用知的ＣＡＩシステム, 情報処理学会論文誌, Vol.26, No.6, pp.1089-1096, 1985.

[14]de Kleer, J.:An assumption-based TMS, Artificial Intelligence, Vol.28, pp.127-162, 1986.

[15]岡本他：知的ＣＡＩのための教授世界知識の表現とその推論の方法, 電子情報通信学会論文誌, Vol.J70-D, No.12, pp.2658-2667, 1987.

[16]Shapiro, E.:Algorithmic program debbuging, MIT Press, 1982.