

ITSのための汎用フレームワーク FITS における 知識ベース構築支援

島崎克也[†] 坂根謙一[†] 野村康雄[†] 太田義一[†] 池田 満[‡] 溝口理一郎[‡]

[†]関西大学工学部 [‡]大阪大学産業科学研究所

あらまし：筆者らは、ITSの本質的構造の明示化と構築容易化を基本姿勢として掲げ、ITSのための汎用フレームワーク *FITS* をこれまでに開発している。一般に、汎用フレームワークを用いたITSの構築には、フレームワーク設計者と教材入力者が携わるが、システムの構築を容易にするには、教材入力者の負担を軽減させる必要がある。そのためには、教材に依存した知識とその表現法を明確にした上でそれらを獲得するための環境を考察しなければならない。本報告では、*FITS*の動作例を示した後、それを実現するための *FITS* における教材依存の知識を明示し、さらにそれらの知識の獲得方法について述べる。

Support of the Knowledge Base Construction for Generic Framework for ITS:FITS

Katsuya SHIMAZAKI[†], Ken'ichi SAKANE[†], Yasuo NOMURA[†],
Yoshikazu OHTA[†], Mitsuru IKEDA[‡] and Riichiro MIZOGUCHI[‡]

Faculty of Engineering, Kansai Univ., †I.S.I.R., Osaka Univ.

Abstract : The objective of our research on the generic framework for ITS is two fold, that is, to make the inherent structure of ITS clear and to reduce the effort required for building an ITS. To attain the objective, the authors have already developed a generic framework for ITS:*FITS*. In general, there are two types of the persons concerned with such a framework. One is the framework developer and the other is the domain expert(teacher). To make system construction easy, we should reduce the domain expert's labor as much as possible. In this paper, after giving some concrete examples of the tutoring discourses generated by *FITS*, we present the requisite knowledge to realize those discourses and discuss the design principle of supporting environment for knowledge base construction.

1 はじめに

ITSの研究目的は学習者への最適な知識伝達手段を確立することにある。この目的を達成するために人工知能、知識工学、認知科学、教育工学などの様々な分野の研究者らがあらゆる観点から考察を深めてきた。筆者らも教材に依存しないITSの構築を目標にシステムの設計を進め、Common-ESPを用いたITSのための汎用フレームワーク *FITS* を既に開発している [MIZOGUCHI90]。

*FITS*の開発において、筆者らは以下の2つの目的達成を念頭に研究を進めた。

- (1) ITSの本質的な構造を明示する
- (2) ITSの構築を容易にする

(1)と(2)は独立ではなく、(1)で得られた構造をフレームワークとして実現することが(2)の達成に貢献する。逆に、ITSの構築を容易にする汎用フレームワークが実現できれば(即ち、(2)の達成)それ自体が(1)の解答になるはずである。つまり、汎用性という切口からITS構築を考察することにより、ITSという問題解決システムの本質的な構造を見定めることができる。筆者らはこのような観点に基づいて研究を進め、教育に固有なタスクと知識構造を明らかにすることにより、人間の教師の持つプリミティブな教育技能に相当するビルディングブロックと教育行動決定機構であるスケジューラを開発した。*FITS*の所有する知識は大きく2つのタイプに分けることができる。一つは教材に依存しない汎用な知識であり、*FITS*が常に保持している知識である。もう一つは教材に依存した知識であり、この知識は必要に応じて獲得される。汎用の知識に関しては別稿 [奥畑89] を参照して頂くことにし、本稿では特に教材依存の知識について述べる。

一般に、ITSには二人の利用者がいる。一人は教育を受ける学習者であり、もう一人は、ITSの実現に必要な知識、特に教材知識の入力を行なう教材の専門家(教師)である。以後、本稿では、前者を学習者と呼び、後者をユーザと呼ぶことにする。また、フレームワークの開発者をインプリメンタと呼ぶことにする。上述した通り、本システムの最大の特徴は、教材依存の知識のみを差し替えることにより原理的にはあらゆる教材に対応できることである。ユーザの入力に対して、インプリメンタは知識の完全性を要求する。つまり、*FITS*が稼働するために最も好ましい構造での知識の入力を要求する。一方ユーザは、知識の入力に対する制約を好まない。このように両者の要求はトレードオフの関係にある。インプリメンタの要求が強くなれば、知識の入力に際してユーザになんらかの制約を課すことになり、余分な負荷を強要することになる。一方、ユーザの要求を単純に受け入

れると、*FITS*の柔軟な動作に支障を来すことになる。従って、知識ベースの構築においてユーザの労力を極力軽減し、かつ知識の完全性を保証できるような環境が必要となる。

以上の観点に基づき、本稿では、*FITS*によって実現される動作例の数例を提示し、それらに必要なメカニズムや教材依存の知識とそれを獲得するためのツールについて述べる。

2 動作例

本章では、ユーザによって入力される知識がどのように役立つかを示すために2つの教材に対する*FITS*の動作例を示し、それぞれについて簡単に説明する。

図1は、穀物の生育問題を教材とした場合のシステムの動作例である。なお、この動作例は考察力の劣る学習者を前提としている。ある土地で穀物が生育するにはその土地の気候、土壌、地形が穀物の生育に適しており、かつ灌漑設備が整っていることが要求される。そして灌漑設備には天然灌漑、または人工灌漑が必要とされ、天然灌漑として適当な降雨があるか、あるいは近くに川があることが条件となっている。この例で*FITS*は、知識の教育順序を考慮して、最初に教育すべき知識として自然灌漑設備に焦点をあて、それに関する学習者の理解状態を確認するところから対話をはじめている(図1(a)~(b))。学習者の理解状態(誤り)の診断を行なったところ、正しくは自然灌漑として降雨と川の2つの選言条件があるべきところを、降雨条件しか認識していないことが判明する。次に*FITS*は学習者の犯した誤りの修正に焦点を変更し、修正教育を開始している。ここでは、学習者の能力(考察力が劣る)を考慮して、学習者がイメージしやすい多数の例題(例えば、カイロは砂漠の国であるが、ナイル川によって農耕が発達した)、それも正解(教材知識)と学習者の理解が一致する例と一致しない例を提示し、それらの例題が持つ属性の違いを考察させることにより、学習者に川に関する条件を想起させようとしている(図1(d))。なお、考察力の優れた学習者を前提とした場合は、提示する例題の数は少なく済み、また特に注意してイメージしやすい例題を提示しなくてもよい。このようにして、自然灌漑、更には灌漑設備に関する教育を終了した後、穀物の生育に焦点を移し、穀物の生育に関する学習者の理解状態を確認している(図1(i))。

次に図2に示す一次方程式を教材とした場合の動作例について説明する。図2では、右辺移項に関する学習者の理解状態を確認するところから対話を始め(図2(a))、学習者が右辺移項の際に必要な符合変換という操作を欠落していることが判明している。ここで、先ほどの2つの例では例題を提示することにより誤りの修正を促そうと試みたが、ここで

はまず、学習者に誤りを認識させ(図2(h)),その後、例題を提示する(図2(i))という二つの段階を経て誤りの修正を行なっている。

以上に示したように、FITSは様々な教材に対応できるだけでなく、学習者の理解状態や教材知識の性質に応じて、教育の焦点や提示する例題の種類などを変更することにより学習者の能力に応じた柔軟な教育を実現している。以下では、上述した動作例のような知的振舞いを実現するための種々の知識について動作例との対応づけを行ないながら述べる。

- (a)新潟に自然灌漑設備はありますか?
Answer ? はい.
- (b)バグダッドに自然灌漑設備はありますか?
Answer ? いいえ.
 バグダッドに川はありますか?
Answer ? はい.
 バグダッドに雨は降りますか?
Answer ? いいえ.
- (c)あなたは今、誤りを一つ犯しています。これからその誤りを修正していきましょう。
- (d)それでは次の質問に考えて下さい。
 以下の地域には自然灌漑設備が整っています。
 バグダッド、カイロ、西安
 以下の地域には自然灌漑設備が整っていません。
 モンゴル、リヤド、クウェート
 これらの属性の違いが分かりますか?
Answer ? はい.
 それでは、以下の説明文の中から正しいと思われるものを選んで下さい。
 (1)バグダッドは海に面しているが、モンゴルは面していない。
 (2)バグダッドには川があるが、モンゴルにはない。
 (3)バグダッドは高緯度に位置しているが、モンゴルは位置していない。
Answer ? (2)
 その通りです。
- (h)それでは引き続き以下の質問に答えて下さい。
- (i)メキシコで米は生育しますか?
Answer ? はい.
 メキシコの気候は米の生育に適していますか?
Answer ? はい.
 メキシコの土壌は米の生育に適していますか?
Answer ? はい.
 メキシコに灌漑設備はありますか?
Answer ? はい.

図1: 対話例(穀物の生育問題)

- (a) $2x+1=3$ の“+1”を右辺移項するとどうなりますか?
Answer ? $2x=3+1$
 ……
- (h)あなたは右辺移項の際に必要な操作を一つ欠落しています。
- (i)それでは、欠落している操作を探しましょう。次の問題に答えて下さい。
 $3-3=0$ の“-3”を右辺移項するとどうなりますか?
Answer ? $3=3$
 その通りです。しかしあなたの考えでは $3=-3$ となるはずですね。
 ……

図2: 対話例(一次方程式)

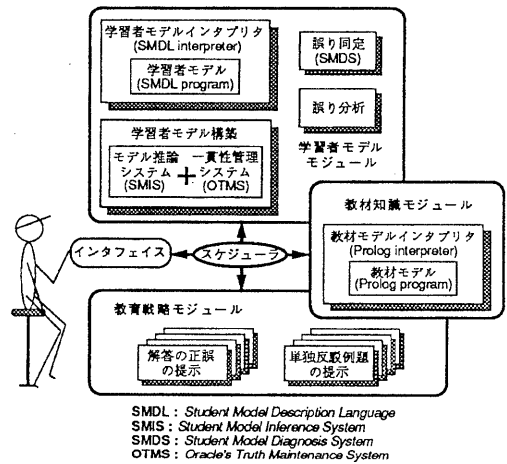


図3: FITSの構成図

3 FITSの構成

本章では、教育に本質的なタスクとその汎用な処理方法の検討を基に開発されたFITSの全体像を紹介する。FITSの構成図を図3に示す。フレームワークの構成要素であるビルディングブロックは、固有のタスクと問題空間を持つ汎用問題解決器として設計されている。以下では各々の機能について概説する。

○学習者モデルインタプリタ (SMDL interpreter) [池田89]: 学習者モデルはSMDL(Student Model Description Language)により記述される。SMDLはPrologの“真(true)”, “偽(false)”という2値の真理値に加え, “さらに, 分からない(unknown)”, “学習者の理解状態を同定できな

い (fail)” という 4 値の真理値を扱うことができる。これにより学習者の理解状態をよりの確に表現することができる。

◦学習者モデル構築 (HSMIS) [池田89]: SMIS は、システムからの質問とそれに対する学習者からの応答を基に学習者の理解状態を帰納推論するアルゴリズムである。

◦誤り同定 (SMDS): SMD (Student Model Diagnosis System) は学習者モデルを診断し、誤答の原因である学習者モデル中のバグ (誤った知識) を同定する

◦誤り分析 [池田87]: 誤りの修正の方向づけを行なうために、誤り同定によって同定された誤りと教材知識との対応づけを行なう。SMDL によって表現される知識の単体は Prolog と同様、クローズの形態となっている。このビルディングブロックでは、クローズ間の距離を最小化するような対応を決定する。

◦教材モデルインタプリタ: 教材知識は Prolog によって表現され、Prolog インタプリタによって解釈される。

◦教育戦略 [島崎91]: 学習者が犯した誤りを修正し、正しい知識を教授するために、ヒントや説明を用いた合計 20 個の教育戦略が用意されている。

◦スケジューラ [奥畑89]: スケジューラは各ビルディングブロックの起動を決定するための枠組である。様々な観点から得られる情報を基に、各状況において最も好ましいビルディングブロックの起動を決定する。

4 FITS の知識ベース

FITS には、様々な教材に対応するためのタスク及び知識が予め用意されており、単に教材依存の知識のみをユーザが入力することによってシステムは動作する。本章では表 1 に示すようなシステムの稼働に必要な知識 (以下、タスク知識と呼ぶことにし、TK (Task Knowledge) と表現する)、並びにシステムに効率のよい動作をさせるために必要な知識 (以下、経験的知識と呼ぶことにし、HK (Heuristic Knowledge) と表現する)、つまりインプリメンタの要求する知識を各ビルディングブロックの観点から整理すると共に、知識表現の具体例を示す。なお、紙面の制限上、複数のビルディングブロックで共有される知識についてはいづれかのビルディングブロックにおいてのみ記述することにする。

4.1 教材知識ベース

FITS では教材知識は Prolog を用いて記述される。図 4 は前述した穀物の生育問題という教材の知識表現例である。grow(X,Y) で始まる節は、場所:Y で穀物:X が生育するには、

```
grow(X,Y):-
    suitable_temperature(X,Y),
    suitable_soil(X,Y),
    suitable_lay(X,Y),
    has_irrigation(Y).
has_irrigation(Y):-
    artificial_irrigation(Y).
has_irrigation(Y):-
    natural_irrigation(Y).
natural_irrigation(Y):-
    has_rain(Y).
natural_irrigation(Y):-
    has_river(Y).
```

図 4: Prolog による教材知識の記述例

Y の気候が X に、Y の土壌が X に、Y の地形が X にそれぞれ適しており、かつ Y に灌漑設備が整っていないことが意味している。

4.2 学習者モデル構築知識ベース

表 1 に示すように、学習者モデル構築には 3 つのタスク知識と 1 つの経験的知識が必要である。ここではこの中から概念ネットワークと節優先探索制御知識について説明する。

<概念ネットワーク (TK)>

SMIS では学習者の理解状態を同定するために、ユーザによって入力された概念の集合とその関係をもとに学習者の持つ知識 (ホーン節に相当) を生成する。この時必要となる知識が以下に述べる概念ネットワークである。概念ネットワークは、ある述語をヘッドとするクローズのボディにつく可能性のある述語を表現しており、概念集合と概念リンクによって規定される。概念集合とは述語の形態と引数の型を規定したものである。概念リンクとはクローズの形態を規定したものである。以下に概念ネットワークの例を示す。

(a) declare(grow(crop,place)).

(b) declare_called(grow(X,Y),[suitable_temperature(X,Y), suitable_soil(X,Y), suitable_lay(Y), ...]).

(a), (b) は、それぞれ概念集合と概念リンクの例を示している。(a) の例は、grow は 2 引数の述語であり、第 1 引数の型は crop、第 2 引数の型は place であることを示している。(b) の例は、第 1 引数をヘッドとするクローズのボディに現れ得る述語を第 2 引数で規定している。

<節優先探索制御知識 (HK)>

システムは、概念ネットワークを基に生成された探索空間

表 1: 汎用フレームワークに必要な知識

汎用タスク	汎用問題解決器	
	タスク知識 (TK)	経験的知識 (HK)
教材知識	Prolog インタプリタ	
	教材知識	—
学習者モデル構築	SMDL+SMIS+ATMS	
	オラクル, 教材知識, 概念ネットワーク	節優先探索制御知識
誤り同定	SMDS	
	学習者モデル, 教材知識	—
誤り分析	節間の類似度計算	
	教材知識, 誤り同定の結果	経験的バグと教材知識との優先的対応
教育戦略	例題生成機構	
	学習者モデル, 教材知識, 教育履歴, 誤り分析の結果,	カリキュラム, 教材知識の組織化, 例題が持つ属性に関する知識
スケジューラ	Soar ベースの問題解決器	
	学習者モデル, 教材知識, 教育履歴, 誤り同定の結果, 誤り分析の結果	カリキュラム, 教材知識の組織化, フレーム知識, ユーザ定義知識, 例題が持つ属性に関する知識,
インタフェイス	対話生成機構	
	テンプレート	フレーム知識

上を探索することにより学習者の理解状態を同定するが、単なる深さ優先の探索は非効率である。本システムでは実際の教育現場の教師が持っていると思われる、特定の教材における学習者の誤りに関する知識を探索の効率化のために用いている。以下に示す例は、学習者が地形に関する条件を欠落するという誤った知識を持ちやすいことを表現している。

```
plausible((grow(T,Crop,Place):-
  suitable_temperature(T1,Crop,Place),
  suitable_soil(T2,Crop,Place),
  has_irrigation(T3,Place)),1)
```

この知識を基に学習者の持ちやすい理解状態を同定するための質問として図 1 の (i) に示す対話が行なわれる。

4.3 誤り分析知識ベース

学習者の持つ誤った知識と教材知識との対応づけは SMDL と Prolog の節間の距離の最小化という手法を用いて行なわれる。この対応づけのためには、教材知識と誤り同定の結果が必須の知識であるが、これに加え、誤り分析モジュールでは効率的な対応づけのための知識を用意している。

<経験的バグと教材知識との優先的対応 (HK)>

SMDL と Prolog の節間の距離の最小化という手法は教材に全く依存しない汎用なものであるが、あまり教育の性質を反映しているとはいえない。そこで、本システムでは、学習者の持つ知識 (学習者モデル) と教材知識との対応づけを効率的に行なうために、経験的に得られる学習者が犯し易い誤りと教材知識とを優先的に対応させるという制御を行なっている。なお、経験的に得られる学習者の犯し易い誤りは学習者モデル構築の枠組で導入されたものを共用している。以下にこの知識の具体例を示す。

```
{ [], (natural_irrigation(Y):-has_river(Y)),
  [(lack_of_model_clause,
    (natural_irrigation(Y):-has_river(Y))) ] }
```

この例は、天然灌漑設備に関する降雨と川という 2 つの選言知識のうちの川の存在を一般的に欠落しやすいという知識を利用したものであり、図 1 でもこの知識を用いて誤り修正の方向付けの効率化を計っている。

4.4 教育戦略知識ベース

学習者に対する効果的な教育を行なうには、様々な観点から得られる情報を分析し、その分析結果に基づいた柔軟な動作が要求される。柔軟な動作は、教育行動として適切なグレインサイズを持った教育戦略を適切に運用することにより実現される。FITS において誤りの修正教育は、20 個の教育戦略を組み合わせることによって実現される。これらの教育戦略を起動するには様々なタスク及び経験的知識が必要である。このうちここでは、例題が持つ属性に関する知識について述べる。

<例題が持つ属性に関する知識 (HK)>

教育戦略は説明を用いたものとヒントを用いたものの 2 つに大別することができる。ヒントを用いた教育戦略の中には例題を用いた教育戦略がある。図 1 の (d) に示した対話文はその中の一つである“差異因子例題の提供”という教育戦略によって生成されたものである。図 1 で示したカイロやバグダッドという例題は、一見、砂漠をイメージする土地であるが、これらは川によって文明が発達した土地として有名である。このようなカイロという例題の属性の著名度を利用することで学習者の思考の支援を容易にすることができ

る。以下に、本システムで整理されている例題の種類を示す。
[著名度] インスタンスの有名さ(知名度など)を表現
[難易度] インスタンスに対する学習者の誤り易さを表現
[重要度] 学習者のインスタンスの理解に対する教師の期待度を表現

これらの知識は次のように用いられる。学習者が誤っている知識を想起し難い場合に著名度の高い例題を適用することで学習者の思考を支援することができる。また適用できる例題が複数存在する場合に重要度の高い例題を優先して適用する。難易度は学習者のレベルと照らし合わせ、能力の優れた学習者には難しい例題を、能力の劣る学習者には優しい例題を選択する。これらの知識表現の一例を以下に示す。
instanceDB(river("カイロ"), yes, [high, middle, middle])
リストは左から順に著名度、難易度、重要度を表現しており、この例では、カイロには川があり、その著名度は特に高いことを意味している。

4.5 スケジューラ知識ベース

スケジューラは *FITS* における多角的視野を持った教育行動決定機構である。教育行動は、様々な観点から得られる情報が If-then 形式の教育行動決定ルール(以下では、スケジューリング知識と呼ぶ)を通して教育行動の嗜好性を半順序関係で表現したグラフへと変換され、そのグラフ上の最大元を選択することによって決定される。この中のスケジューリング知識には、教育行動を決定づけるための様々な知識が準備されている(表 1 参照)。その大半は、教材に依存しない知識であり、例えば以下のような知識がある。

If(教材 A に関して学習者モデルを構築している),
(教材 A に関する学習者モデルの信頼度は低い),
then(教材 A に関する学習者モデル構築の続行).

図 1 の (a)~(b) ではこの知識が適用されている。まずはじめに新潟の自然灌漑について尋ねたところ、学習者が正解を示したことから学習者が自然灌漑に関して正しい知識を持っているとみなせるが、十分な情報に基づく結論ではないので信頼性に欠ける。このためバグダッドについて同じ質問を行なっている。現在のところ教材に依存した知識は少ないが、教育行動決定のための貴重なファクタであり、必要に応じて随時設定する。以下では、教材に依存したスケジューリング知識について述べる。

<教育カリキュラム (TK)>

一般に、知識の習得に際しては知識間の優先順位が存在する。例えば、図 1 に示したように、穀物の生育問題では、灌漑設備に関する知識の習得は、穀物の生育に関する知識の習得に優先する。また方程式に関する知識の習得には、乗算や加算などの四則演算の習得が優先する。もし知識間の先行

関係が明示されていなければ、学習者がなぜ誤りを犯すかをシステムは認識できないおそれがある。従って、どの知識がどの知識に先行するののかという知識間の先行順序をスケジューリング知識として明記する必要がある。スケジューラはこの知識を元に教育カリキュラムの作成を行なう。

<ユーザ定義知識 (HK)>

FITS では、ユーザが独自のプログラムを組むことが可能であり、それを適宜起動する知識(ルール)を定義することができる。このようなルールには以下に示すものがある。

If(例題の属性を想起させるための教育戦略を適用した)
(学習者は例題の属性を想起できない)
then(例題の属性を読みとり易いグラフの提示).

例えば、このルールは穀物の生育条件の教授において学習者が気候条件を想起できない場合に、提示した例題の気候に関するグラフを提示するルールである。このようなルールの記述を認めることにより、ユーザの意思を反映させたシステムの動作が可能となる。

<教材知識の性質を反映した教育戦略 (HK)>

筆者らが用意した教育戦略の中には、教材知識の性質により学習者に与える教育効果が異なるものが存在する。そこで筆者らは、これまでに教育戦略の教育効果に影響を及ぼす教材知識の性質を分析し、教材知識を分類した [奥畑90]。筆者らは、宣言的/手続き的、概念の抽象度、概念間の形式的操作、探索空間の広さという4つの教材知識分類のため観点を提案している。第2章では2つの異なる教材において、知識を一般化している場合に適用する教育戦略がどのように異なるかを示した。例えば穀物の生育問題では、最初に“差異因子例題の提供”という教育戦略を用いて誤りの修正を促している(図 1(d))。一方、一次方程式の例では、まず“誤りの種類の提示”という教育戦略を用いて学習者に誤りに関する認識を明確にさせた(図 2(h))後で誤りの修正を促している。この違いは宣言的/手続き的という観点からの教材知識の分析・分類を反映したものである。このように対象とする教材を4つの観点から分類し、その結果を踏まえた教育戦略の運用が行なわれる。ここでの知識表現例を以下に示す。

*domainClassification(growthOfCrop,
[declared, concrete, empty, narrow])*

この例は、穀物の生育という教材は宣言的であり、概念の抽象性が低く、概念間の形式的操作が定義されておらず、探索空間が狭いということを表している。

4.6 インタフェイス知識ベース

インタフェイスは、ITSのみならず、システムと人間との間の情報伝達を要求するマンマシンシステムにおいて大変重要な機構であるが、自然言語理解など、このモジュールの

開発は困難な状況にある。本システムではこのモジュールは開発途中の段階にあるが、現在の枠組にフレーム機能などを取り込むことによってより効果的な教育が実現できると考えている。以下に本システムの対話生成において必要なテンプレートとフレーム知識について述べる。

<テンプレート (TK)>

本システムでは現在、学習者との意思伝達をキーボード及びディスプレイを通して行っている。質問や説明、ヒントなどの学習者への出力は対話生成部において生成され、ディスプレイ上に提示される。対話生成部では各ビルディングブロックから送られた情報とテンプレートを基に対話の生成を行う。テンプレートは以下のように表現されている。

```
ifKB(+grow(X,Y),[Y,"で",X,"は生育します。"]);  
ifKB(-grow(X,Y),[Y,"で",X,"は生育しません。"]);  
ifKB(?grow(X,Y),[Y,"で",X,"は生育しますか?"]);
```

述語 ($grow(X,Y)$) の前の +, -, ? はそれぞれ肯定, 否定, 疑問を意味しており、図 1 の (i) の対話文はこの中の 3 番目の知識を用いて生成されたものである。

<フレーム機能 (HK)>

本システムでは、フレーム機能を準備することにより、学習者の思考の誘導を簡単化することができる。例えば、本システムの対象とする教材の一つである酸と塩の反応の教育場面では、図 1 で示した学習者の思考に対する反例を提示するという方法以外に、フレーム機能を用いて実験環境を設定し、その上で反応の成立の可否を伝授することができる。このようなフレーム機能を用意することによって学習者との知識伝達がよりスムーズになる。またフレーム機能は、学習者の持つ認知モデルをグラフィカルにディスプレイ上に再現し、その上で学習者とのインタラクションをとりながら教育を進めることを可能にする能力を有している。

5 知識ベース構築支援

これまで汎用フレームワークに必要な知識について述べてきた。ここで、第 1 章で述べた筆者らの ITS 開発における基本理念を思い起こして頂きたい。筆者らは ITS の本質的構造の明示化及び構築容易化を目標にこれまで考察を続けてきた。ITS の構築容易化は、ITS の本質的構造をフレームワークとして実現するインプリメンタとそれをチューニングするユーザ両者の要求を同時に満たしてはじめて実現されたと言える。本章では、“フレームワークの内部構造を意識しないで教材知識を入力したい” というユーザの要求を満足させるために、まず第 4 章で述べた様々な知識をユーザの視点から整理した上で、それらの知識の入力を支援するツール群について述べる。

5.1 ユーザ入力知識

ユーザによって入力される知識は、表 2 に示すように専門知識と学習者の理解に関する知識の 2 つに大別することができる。専門知識には教材知識、インタフェイス知識という 2 つのタスク知識と、例題の属性、カリキュラム、教材の種類、スケジューリング知識という 4 つの経験的 (付加的) 知識がある。また、学習者の理解に関するタスク知識には概念ネットワークがある。以下でユーザが入力すべき知識をまとめ、ユーザの立場から見た簡単な説明を加えておく。

<教材知識>

学習者に教育すべき知識 (Prolog の形態)

<インタフェイス知識>

学習者に提示する対話文を生成するための知識

<例題の属性>

例題が持つ属性に関する知識であり、例の有名さや、特に教授したい例であるか、学習者が誤りやすい例かを表現した知識

<カリキュラム>

学習者に教える教材の順番に関する知識

<教材の種類>

教材知識の性質を明記した知識

<スケジューリング知識>

ユーザが経験的に得た効果的な教育行動系列を設定するための知識

<概念ネットワーク>

学習者が持ち易い理解を表現した知識

以上に示した知識のみをユーザが入力するだけで、表 1 に示した各ビルディングブロックごとの汎用フレームワークに必要な知識が構築され、図 1 や図 2 に示したような対話を実現される。

5.2 知識ベース構築支援ツール

前節で述べた知識のユーザによる容易な入力を実現するために、本システムでは概念エディタ、クローズエディタ、インスタンスエディタ、教材分類エディタ、カリキュラムエディタ、スケジューリング知識エディタなど、多種多様なツールを用意する。ここでは、これらのツールの中から、各タスク毎に整理された知識の獲得に貢献する概念エディタとクローズエディタについて述べる。なお、FITS が対象とするユーザは Prolog に関する基本的知識を備えたものとする。

基本的にユーザは概念エディタとクローズエディタを用いて知識を定義する。概念エディタとは述語の形態 (述語とそれが持つ変数の型) を入力するためのツールである。クローズエディタは概念エディタによって記述された述語を

表 2: ユーザ入力知識

	タスク知識	経験(付加的)知識
専門知識	教材知識, インタフェース知識	例題の属性, カリキュラム, 教材の種類, スケジュール知識
学習者の理解に関する知識	概念ネットワーク	—

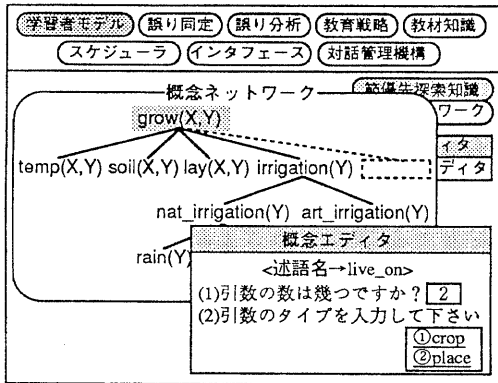


図 5: 概念ネットワークの作成

節の形に変換するツールである。この2つのエディタを用いて教材知識の作成が終了すると、学習者モデル構築のための概念ネットワークの作成を行なう。図5は概念ネットワーク作成中の様子を示したものである。まず、図に示すような正しい知識に関する概念ネットワークがスクリーンに現れる。ここでまず、提示された述語に関して学習者が欠落しやすいと思われる述語にフラグを立てる。これは第4.2節で述べた節の優先探索に用いられる。次にユーザに学習者が付加しやすい知識、すなわちヘッド(この場合は grow(X,Y))のボディにつく可能性のある述語が存在するかどうかを尋ねる。ここでスクリーンに提示されている以外に存在する場合には、そのヘッドをマウスでクリックする。するとシステムはその述語名をたずねる。そして入力された述語名が既に定義されているかどうかをシステムが判断し、未定義の場合には、概念エディタを用いてユーザに述語の形態を明記させる。この例では、学習者がある土地で米が生育する条件として考え得る“その土地の人々が米を主食としている”という条件を入力している。

6 おわりに

本稿では、FITSを用いて ITS を構築する際に必要となる教材依存の知識について整理し、さらにそれらをユーザの観点から再整理し、知識ベース構築のためのツールについて述

べた。本システムでは、概念エディタ、クローズエディタなどのごく単純なツールを用いて、インプリメンタが要求する多様な教材依存の知識をシステムの知識構造をユーザに意識させることなく獲得することができる。

しかし、さらにシステムの性能をあげるためには本稿で述べたフレーム機能に用いるグラフィック知識などの支援環境の充実や、獲得された知識の管理、拡張などに関する考察が必要である。

謝辞:本研究の一部は文部省科学研究費(重点領域研究「CAIによる知識獲得」, No. 03245106)の援助を受けた。

参考文献

- [奥畑89] 奥畑健司, 池田満, 野村康雄, 溝口理一郎, “高度個別教育のための教授知識とそのスケジューリング機構”, 電子情報通信学会技術研究報告, ET89-12, AI89-12, pp. 77-84, 4 1989.
- [奥畑90] 奥畑健司, 坂根謙一, 島崎克也, 太田義一, 野村康雄, 溝口理一郎, “教育戦略の観点から見た教材知識の分類と組織化について”, 人工知能学会研究会資料, SIG-FAI-HICG-KBS-9001, pp. 67-76, 12 1990.
- [池田87] 池田満, 鈴木信夫, 溝口理一郎, 山口高平, 角所収, “知的 CAI のためのフレームワークの検討”, 情報処理学会シンポジウム論文集, 87, No.1., pp. 29-38, 1987.
- [池田89] 池田満, 溝口理一郎, 角所収, “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, Vol.J72-D, N.1, pp. 112-120, 1 1989.
- [島崎91] 島崎克也, 坂根謙一, 奥畑健司, 太田義一, 野村康雄, 溝口理一郎, “教育目的の整理とそれに基づく教育戦略の運用について”, 人工知能学会研究会資料, SIG-KBS-9101, pp. 1-10, 4 1991.
- [MIZOGUCHI90] MIZOGUCHI, Riichiro and IKEDA, Mitsuru, “A Generic Framework for ITS And Its Evaluation”, In *Proceedings of ARCE*, pp. 303-312, 1990.