

## 人に教える行為により階層構造を持つ問題の 解法を学ぶための個人学習支援システム

澤田石 礼秀<sup>†1</sup> 山本 景子<sup>†1</sup> 倉本 到<sup>†1</sup>  
辻野 嘉宏<sup>†1</sup> 水口 充<sup>†2</sup>

数学の問題に代表される階層構造を持つ問題を解くのに必要なことは、(1) 問題を部分問題に分割できることと (2) 部分問題が解けることである。このような問題を解く手順を段階的に説明させるために従来の LBT システムを適用すると、問題分割を単純に教えられないという問題がある。本研究では、この階層構造を持つ問題を対象とした、個人で LBT に基づく学習を行うことを実現するシステムを提案する。このシステムでは、問題の分割を直接ユーザに文や式を入力させる形式で行う。また、これを正しく理解できていないユーザに分割に注力させるようなヒントの提示を行う機能を有する。評価の結果、本システムは、ユーザに、「エージェントに教えている」という感覚を与えており、提案システムにより解いた問題だけでなく提案システムで解いていない形式の問題に対しても、それに合わせて知識を適用できるようになる可能性が示された。

### Self-Learning Support System with “Learning by Teaching” Method for Learning Solutions of Hierarchical Problems

NORIHIDE SAWADAISHI,<sup>†1</sup> KEIKO YAMAMOTO,<sup>†1</sup>  
ITARU KURAMOTO,<sup>†1</sup> YOSHIHIRO TSUJINO<sup>†1</sup>  
and MITSURU MINAKUCHI<sup>†2</sup>

The process of solutions to hierarchical math problems consists of following two steps: 1. understanding elemental knowledges and 2. dividing a problem into subproblems to apply the knowledges. LBT (learning by teaching) system does not simply support such problems. We propose a LBT based self-learning system for learning solutions of hierarchical problems. In this system, a user inputs sentences or formulas to divide problem, and the system gives hints to divide problem to user who does not understand the step 2. As a result of

experimental evaluation, this system can produce the sense of “I taught to the agent”, and some of the participants can answer not only the similar questions which they taught to the agent but questions which are a little different from the similar one.

#### 1. はじめに

数学の問題に代表される階層構造を持つ問題を解くためには、一般にその問題を部分問題に分割し、それぞれの答えを求め、統合するという手順を踏む。このような問題の解法を学習する方法として、一般にはドリル形式の学習が用いられる。しかし、ドリル形式の学習には解答は載っているがその解答にたどりつく道筋が説明されないことが多い。そのため、問題に適用すべき知識とその実際の適用法が学習できない。このことから、ドリル形式の学習では、知識は習得できても解法が学びにくいと考えられる。

そこで、LBT (Learning By Teaching)<sup>1)</sup> と呼ばれる学習手法を用いる。LBT は人に教えるという行為を通じて、順を追って解き方を説明していく学習法である。LBT では、学習者が他者に教えた内容を、他者の反応から自己にフィードバックすることによって、自らの知識について内省し、より深く学ぶことができる。このことから、LBT を用いることで、ドリル学習では学びにくい解法の学習が行えると考えられる。

個人学習を支援するため、LBT を計算機上に実装した研究はいくつか見られるが、これを数学的な階層構造を持つ問題に適用した例はみられない。そこで本論文では、個人学習において、階層構造を持つ問題に LBT を適用する手法を提案する。

#### 2. 階層構造を持つ問題の学習

##### 2.1 階層構造を持つ問題の特徴

数学の問題に代表される階層構造を持つ問題は、それより小さい問題（部分問題）の組み合わせとして構成されている。図 1 に示すように、例えば「太郎君が家から駅まで、初め分速 70m で歩き、途中から分速 300m で走りました。太郎くんが歩いた時間は 30 分、走った時間は 6 分でした。家から駅までの距離は何 m でしょう？」という問題は、「家から駅ま

<sup>†1</sup> 京都工芸繊維大学  
Kyoto Institute of Technology

<sup>†2</sup> 京都産業大学  
Kyoto Sangyo University

問題:  
太郎君が家から駅まで、初め分速70mで歩き、途中から分速300mで走りました。  
太郎くんが歩いた時間は30分、走った時間は6分でした。  
家から駅までの距離は何mでしょう？

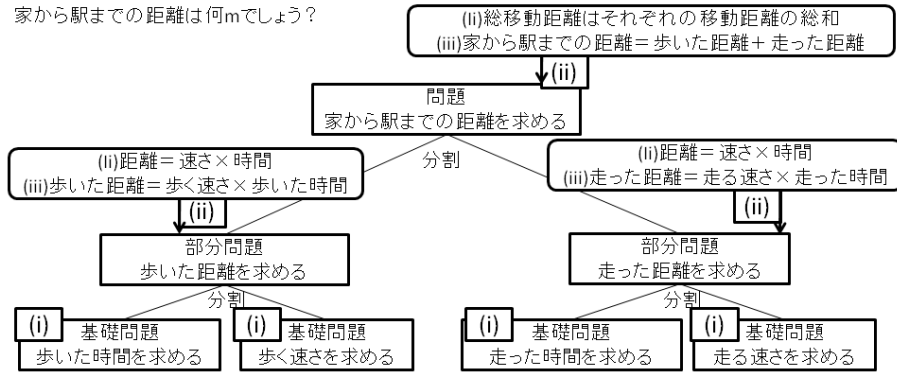


図 1 階層構造を持つ問題  
Fig. 1 Hierarchical Problem.

での距離 = 歩いた距離 + 走った距離」という関係から、「家から駅までの距離を求める問題」は、「歩いた距離を求める問題」と「走った距離を求める問題」という部分問題から構成されていることがわかる。さらに、「歩いた距離を求める問題」も、「距離 = 速さ × 時間」という関係から、「歩いた距離を求める問題」は、「歩く速さを求める問題」と「歩いた時間を求める問題」という部分問題から構成されていることがわかる。

このような階層構造を持つ問題を解く一般的な手順は、

- (1) 問題を分割するための知識を適用し、問題を分割する
- (2) 分割を繰り返して、直ちに解決可能な基礎的な部分問題（基礎問題）にして解く
- (3) 解かれた部分問題を統合して最終的な問題の答えを求める

である。つまり、階層構造を持つ問題を解くときには、以下の3つが必要となる。

- (i) 基礎問題を解くために使う知識（基礎知識）
- (ii) 問題を分割する方法

分割に必要な公式を覚えていること、そして、その公式を与えられた問題を解くために用いること。例：「距離 = 速さ × 時間」：「歩いた距離を求める問題」には「距離 = 速さ × 時間」という公式を使うことで、「速さ」と「時間」の問題に分ける

- (iii) 問題の具体化

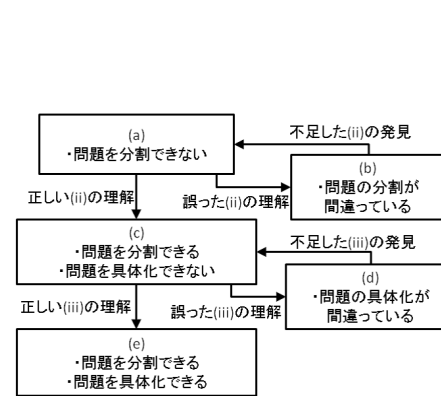


図 2 学習者の部分問題への分割における状況の遷移  
Fig. 2 Learner's Status of Solving Hierarchical Problem.

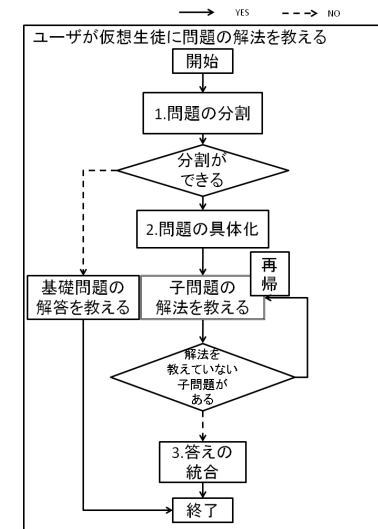


図 3 LBT を用いて問題を解くときのフロー  
Fig. 3 Flow of Solving a Problem with LBT.

分割された問題を実際の問題に適用し、解釈すること。例：「距離」、「速さ」、「時間」がそれぞれ、「歩いた距離」、「歩く速さ」、「歩いた時間」となることから、「歩いた距離 = 歩く速さ × 歩いた時間」と解釈すること

ここで、(ii) (iii) を理解しているかどうかに着目して、学習者が問題の解き方を学習するときの学習者の状態の遷移は図 2 のようになる。

## 2.2 Learning By Teaching

「人に教えるという行為によってなされる学習」には、次の学習が含まれている。まず、学習者はある学習内容を他者に教えるために、教える内容に関する自身の知識を再確認する。このとき、知識の不足や誤りに気づくことができれば、学習者は「知識が足りない」ということを学習する。また、教える行為は相手に理解させるという意識のもと行われる。相手に理解してもらうためには話の繋がりがわかるように順序立てて話を進めなければならない。そのため教える行為においては必然的に問題を段階的に解くことになる。

LBT を用いた教育支援システムについての研究として、Leelawong ら<sup>2)</sup> の研究がある。Leelawong らの“The Betty’s Brain System”において、学習者は擬人化エージェントであ

る Betty に概念ネットワーク, 例えば川辺の生態系や窒素の循環などについて教える. 学習者は, Betty に自身が教えた内容について質問し, その回答を見ることで教えた内容の正誤を認識し, 誤りを正す. このように, 学習者は LBT を用いて概念ネットワークを学習する. このシステムは, 概念と概念との関係についての知識を学ぶものである.

しかし, 階層構造を持つ問題の解法には, 基礎知識や公式といった知識を覚えていることだけでなく, 問題を分割し具体化するという, 問題を部分問題に分割することが含まれる. そのため, 概念や知識を単位とする LBT システムでは, ある知識の一部である問題分割および解釈だけを個別に問うことが難しい. また, それを理解していることを確認するためには, 単位知識 (基礎知識) の単純な羅列ではなく, それらの間の構造を全て学習者の手で構築される手続きが必須となる. つまり, これら (ii) (iii) を文章や式を直接記述する形式で教える LBT システムが必要となると考えられる.

なお, 教える側 (学習者) の知識の不足と理解不足によって学習が進まなくなる場面がある. 普通, 学習システムはこれらの知識やその理解を学習させるのが目的である. そのため, 知識獲得と知識の扱い方の理解を促す方法が必要である. このとき, 持っていない知識の獲得を促すにはその知識の内容を示せばよい. しかし, 特に (ii) (iii) の学習にあたる知識の扱い方の理解は, 扱い方を自力で構築することによってのみ学習が進むので, 単に扱い方を示しても学習にならない. そのため, システムは「知識の扱い方」にのみに注力できるようなヒントを与える必要がある.

### 2.3 LBT の適用

LBT を階層構造を持つ問題に適用するためには, 学習者が「教える」行為をする対象となる生徒役が必要である. 本研究では, LBT を個人で行えることを目指し, 計算機上に実現されたエージェントが生徒役を行う.

LBT を用いて (ii) および (iii) を学ぶフローは図 3 のようになる. 以降, 生徒役であるエージェントを「仮想生徒」, 「学習者」をユーザと呼ぶ. ユーザは, システムが用意した語の中から必要な語を選択し, それを自由に並べることで文章や式を作成する. 作成した文章や式をシステムに入力することが, 仮想生徒に教えたことになる. 以下では, 例として「問題: 分速 70m で 30 分歩いた. 歩いた距離を求めよ」に適用した場合と共に述べる.

#### 1. 問題の分割

図 2 における (a) から (c) に移るために (ii) 問題の分割をユーザに理解させる. ユーザはまず, 適用する知識と, それにより問題がどのように分割されるかを, システムに文章や式で入力することで, 仮想生徒に教える.

教えた知識が間違っている, または, 教えた知識は正しいが, ここで適用する知識でない場合, 仮想生徒は教えてもらった知識では問題が解けないことをユーザに伝える. ユーザは仮想生徒が問題を解けないことを認識することで, 自身の誤りに気づく. つまり, 自身の知識が間違っていることを理解する. これは図 2 における状況 (b) から (a) へ遷移したことを示す.

例題の場合, ユーザは「歩いた距離を求める問題」に「距離・速さ・時間の関係式」を適用すること, そして, その問題が「距離=速さ×時間」という関係式により 2 つに分割されることを教える. この例題の場合, 無関係な知識, 例えば「濃度の関係式」を適用することをユーザが教えた場合, 仮想生徒はその知識を使って問題が解けないことをユーザに伝える. これによりユーザは「濃度の関係式」がこの問題には無関係であることを知る. そこで, ユーザは適用する知識がわからないことに気づく.

教えた知識の内容と名称が正しく, かつ, 適用すべき知識である場合, 2 に進む. これは図 2 における状況 (c) に遷移したことを示す.

#### 2. 問題の具体化

図 2 における (c) から (e) に移るために (iii) 問題の具体化をユーザに理解させる. ユーザは仮想生徒に, (ii) のときに使った公式を問題に合わせて具体化した文章や式を, システムに入力することで教える. 具体化が正しい場合, できあがった各要素は部分問題になっている. そこで, これらの部分問題に対してこのフローを再帰的に繰り返す. このとき図 2 における (c) から (e) へ遷移したことを示す. 具体化の結果が誤っていた場合, 仮想生徒は部分問題の分割過程を理解できないことをユーザに伝える. ユーザは教えた内容を仮想生徒が理解できないことを認識することで, 自身の誤りに気づく. これは図 2 における (d) から (c) へ遷移したことを示す.

例題の場合, ユーザは「歩いた距離を求める問題」に「距離=速さ×時間」を適用すると, 「歩いた距離=歩く速さ×歩いた時間」になることを教える. そして, 「歩く速さを求める問題」と「歩いた時間を求める」という部分問題に分割できることを教え, それぞれの部分問題を同じ手順で教える.

#### 3. 答えの統合

仮想生徒が部分問題を全て解き終わると, 仮想生徒は答えを統合することで問題の答えを導く. ユーザは仮想生徒の解答を見ることで, 自身の知識とその適用が正しくなされたことを確認する.

例題の場合, 「歩く速さ」と「歩いた時間」の答えがわかると, 「歩いた距離=歩く速

さ×歩いた時間」から、仮想生徒は「歩いた距離」の答えを求めることができる。「歩く速さ＝分速 70m」,「歩いた時間＝ 30 分」であることから、仮想生徒は「歩いた距離＝ $70 \times 30 = 2100\text{m}$ 」と「歩いた距離を求める問題」の答えを示す。

#### 2.4 ヒント

ユーザが仮想生徒に一方的に教えることだけを実現すると、ユーザが自分自身の力だけで教える内容を思いつけない場合、フローを先に進めることができず、ユーザの学習が進まなくなる。このとき、2.2 で述べたように単純に理解できていない知識を直接与えることはできない。そこで、下記に示すようなユーザが教える内容を思いつききっかけとなるようにヒントを与えることで学習者の理解を支援する。

##### 2.4.1 (ii) 問題の分割の理解支援のためのヒント

(ii) 問題の分割を理解するためには、使う知識がどのように構造化されていて、どのように分割できるか、という点を示さずに、使う知識を思い出させる必要がある。このとき、分割における各部分問題を意味する公式の各項の具体的な名称は重要ではない。そこで、ユーザが使う知識を思い出さきっかけとなるように、文を構成するためにシステムが用意した語のうち、どれを必要とするか、または、しないかをヒントとして与える。

例えば、「距離と時間と速さの関係式」という分割知識を教える場合では、関係のないキーワードのヒントとして「「時速」は関係ない」や、関係のあるキーワードのヒントとして「「距離」は関係ある」とユーザに情報を与える。これによりユーザに、「距離＝時間×速さ」という教える式を作るためのキーワードの想起を促す。

##### 2.4.2 (iii) 問題の具体化の理解支援のためのヒント

知識を問題に適用する場合において、ユーザは、(ii) 問題の分割から問題の構造が得られている。このことから、(iii) 問題の具体化を理解してもらいたい場合には、ここまでで示されている問題の構造で悩まないように、問題の構造をユーザに与える。そうすることで、(iii) 問題の具体化にユーザを集中させることができる。

例えば、「距離・速さ・時間の関係式」が「距離＝速さ×時間」とユーザが覚えていることとする。「歩いた距離を求める問題」の問題構造は「距離＝速さ×時間」から、「○○の距離＝○○の速さ×○○の時間」ということを理解している。この構造を式や文を作るヒントとしてユーザに与えることで、式や文をどのように表記するかで混乱することはなくなる。結果、(iii) 問題の具体化にユーザが集中するようになる。

### 3. LBT による個人学習支援システム

ここでは、2. で述べた手法に基づく LBT による個人学習支援システムの実装について述べる。本システムは、操作部とノート部の 2 つのインタフェースを有している。まず、3.1 では本システムで用いる学習内容のデータ構造について述べ、3.2 では操作部とシステムの動作を、3.3 でノート部について述べる。

#### 3.1 データ構造

本システムで使用するデータは、問題の分割に使われうる知識を表現する知識データと問題を表現する問題データ、問題の分割と具体化、答えの統合を表現する解き方データと、教える文章を作成するための用語を表現する用語データの 4 種のデータからなる。データ間の関係を図 4 に示す。

用語データは、知識データにおける、知識の名称・内容、および、問題データにおける、問題の解答、および、解き方データにおける、部分問題を導出するための解答、の文章や式をユーザが作成するために使用されうる単語、記号、数字である。以下では用語データは「」書きで表現する。例：「距離」「速さ」「＝」「×」「の」「80」

知識データは、以下の 3 つのデータからなる。

- 知識の名称（用語データ）  
例：「距離・速さ・時間の関係式」
- 知識の内容（用語データ列）  
例：「距離」「＝」「速さ」「×」「時間」
- 問題の分割のためのヒント（用語データ集合）  
例：「距離」「速さ」「時間」「＝」「×」  
また、「問題データ」は、2 つのデータからなる。
- 問題文（文字列）  
例：歩いた距離を求めよ
- 問題の答え（用語データ）  
例：「2100m」
- 自身の問題の解き方群（解き方データ集合）

(ii) 問題の分割の仕方によって解き方は複数考えられる。そのため、1 つの問題に対して解き方は複数存在する。

解き方データは以下のデータからなる。

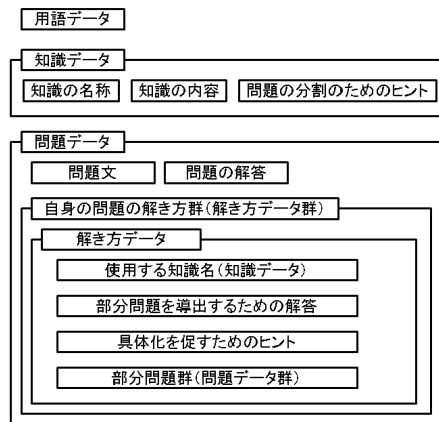


図 4 データ構造  
Fig. 4 Data Structure.

- 使用する知識 (知識データ)  
例: 距離・速さ・時間の関係式
- 部分問題を導出するための解答 (用語データ列)  
例: 「歩いた」「距離」「＝」「歩く」「速さ」「×」「歩いた」「時間」
- 問題の具体化のためのヒント (用語データ列)  
例: 「○○」「の」「距離」「＝」「○○」「の」「速さ」「×」「○○」「の」「時間」
- 部分問題群 (問題データ集合)  
例: 歩く速さの問題データ, 歩いた時間の問題データ

### 3.2 操作部

操作部 (図 5) では, 上から順に, 問題文, 仮想生徒のセリフ, 語群, ユーザの発言の入力欄, ユーザの発言の履歴が表示される。

ユーザは仮想生徒に教えるために式や文 (以降, 発言文) を作る。発言文の作成方法を完全にユーザの自由にしてしまうと, 文章が多様多様になり, システムが正解と比べることが困難となる。そこで, 本システムでは発言文を作るために使える語を用語データ内の語のみに制限した。

ユーザは語群から言葉を選んで並べることで発言文を作り, 「だよ」 ボタンを押すことで発言する。発言文を作り直すときは「じゃなくて」 ボタンを押す。仮想生徒の質問に答えら

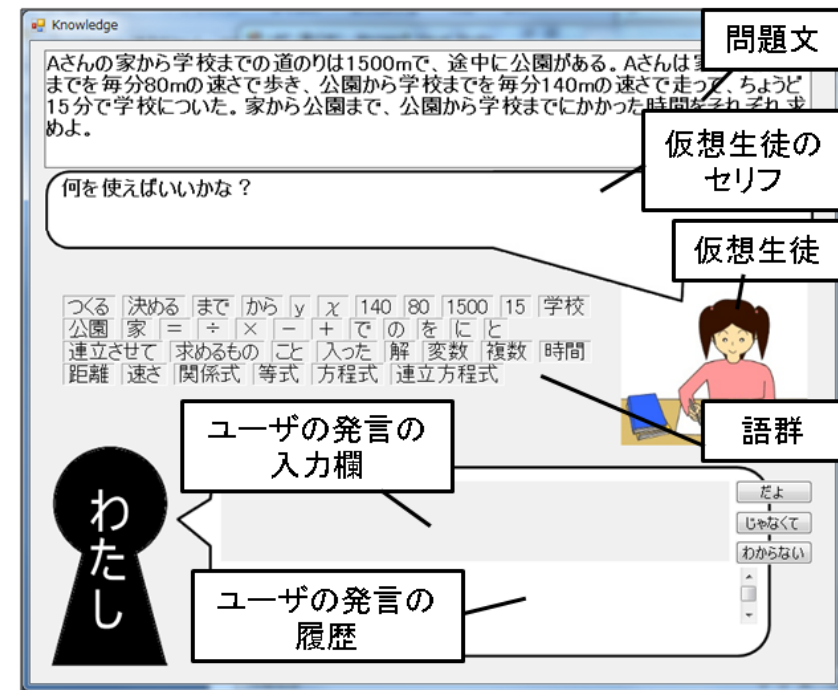


図 5 操作部  
Fig. 5 Interface of The Proposed System.

れないときは「わからない」 ボタンを押す。

以降, 2.3 における, 問題の分割, 問題の具体化, 答えの統合の段階をそれぞれ「問題分割フェーズ」, 「問題具体化フェーズ」, 「解答統合フェーズ」として実装した。仮想生徒の質問内容とボタンを押した後の動きを, それぞれのフェーズごとに説明する。

#### 3.2.1 問題分割フェーズ

問題分割フェーズでは, (ii) 問題の分割を行うために, ユーザが持っている知識の再確認と適用を行う (図 6)。

仮想生徒はユーザに対して, その問題に適用する知識について尋ねる。仮想生徒はまず名称について尋ねる。

ここで「わからない」 ボタンを押した場合, 仮想生徒が知識の名称を思い出し, その知識

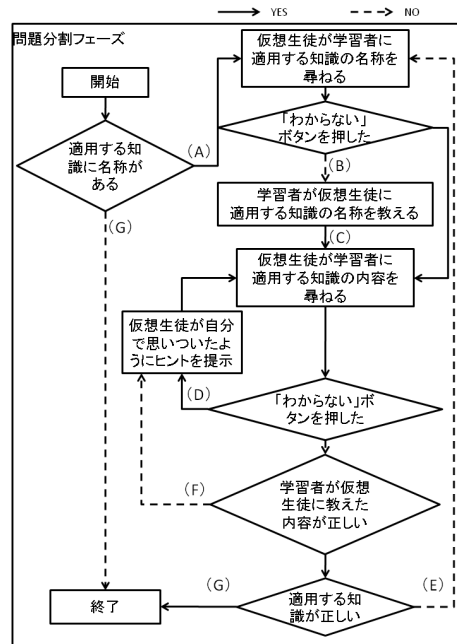


図 6 問題分割フェーズ  
Fig. 6 Problem Division Phase.

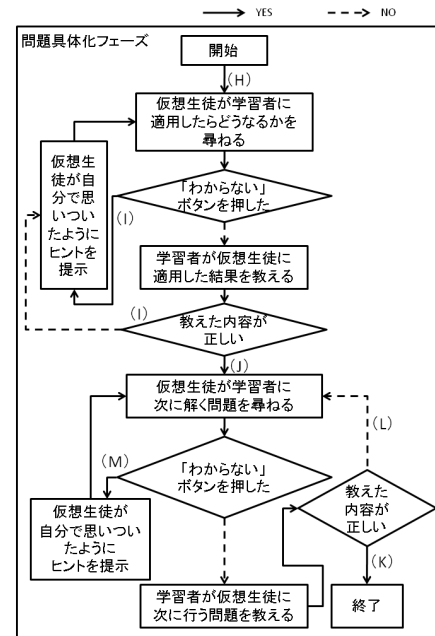


図 7 問題具体化フェーズ  
Fig. 7 Subproblem Making Phase.

の内容についてユーザに尋ねる。

ユーザの知識の名称の発言があった場合、次に、その知識の内容について尋ねる。ここで「わからない」ボタンを押した場合、仮想生徒は「知識データ」にある「(ii) 問題の分割を理解させるためのヒント」を提示する。知識の内容を正しく答えていない場合、仮想生徒がユーザに間違っていることを指摘し、適用する知識の内容を再び尋ねる。ユーザの教えた内容が「解き方データ」の使用知識に含まれていた場合、問題具体化フェーズに進む。

### 3.2.2 問題具体化フェーズ

問題具体化フェーズでは、(iii) 問題の具体化を行い、部分問題を導く (図 7)。

仮想生徒はユーザに対して、知識を適用すると問題はどうか具体化されるのかを尋ねる。ユーザの教えた内容が「解き方データ」の「部分問題を導出するための解答」と同じだった場合、仮想生徒はユーザに対して、次に解く問題を尋ね、指定された部分問題に対してこの

プロセスを繰り返す。すべての部分問題が解かれたら解答統合フェーズに進む。間違っている、もしくは、「わからない」ボタンを押した場合、仮想生徒が (iii) 問題の具体化を理解させるためのヒントを提示する。

### 3.2.3 解答統合フェーズ

解答統合フェーズでは、答えが導かれている部分問題を統合し、その部分問題により構成されている問題の答えを導く。導き出された答えを、仮想生徒はユーザに確認させる。統合ができなくなる、つまり元々の問題の解答が出るまで、解答統合フェーズを繰り返す。

### 3.3 ノート部

ノート部には、ユーザが今何をしているか、仮想生徒にこれまで何を教えたかを理解するための情報を提示する。情報は、問題文と部分問題を導くための解答、次に解く部分問題、問題の答えである。これらを表示された順に自動的にノート部に提示する。

## 4. 評価実験

### 4.1 目的

提案システムが、学習を支援できるかを検証する実験を行う。このとき、学習効果に加えて LBT による効果があったかを確認するために、ユーザが教えている感覚を与えられたかについてのアンケート調査を行う。

### 4.2 方法

3. で述べた提案システムを用いて実験を行った。学習する階層構造を持つ問題として、連立方程式の「距離と速さからかかった時間を求める」文章題を作成した。被験者は、中学 2 年生の 24 人である。

まず初めに、被験者に対して、その時点で学習内容をどの程度理解できているかを把握するためのプレテストを行った。プレテストの内容は、次の連立方程式の文章題 4 問である。問題の難易度は次の 4 段階とした。

- レベル 1: システムを使って学習する問題と数値を変えただけのもの
- レベル 2: 学習する問題と使う知識は同じだが求める解の種類が違うもの
- レベル 3: 距離の問題だが学習する問題と問題構造が違うもの
- レベル 4: 連立方程式の「食塩水の濃度を求める」問題

レベル 1~3 と違い、レベル 4 はシステムを使用しても全ての段階における (ii) (iii) が学習されない「食塩水の濃度の関係式」を使う問題である。実際に使用した問題を表 1 に示す。次に、提案システムの操作方法を理解するために、練習問題を 1 問分、実験者の指示通り

表 1 学習する問題とプレテストの問題

Table 1 Target Problem in This System and Problems of Pre-Test.

問題の種類	問題文
学習する問題	Aさんの家から学校までの道のりは1500mで、途中に公園がある。Aさんは家から公園までを毎分80mの速さで歩き、公園から学校までを毎分140mの速さで走って、ちょうど15分で学校についた。家から公園まで、公園から学校までにかかった時間をそれぞれ求めよ。
プレテスト問題 1	Aさんの家から学校までの道のりは1800mで、途中に公園がある。Aさんは家から公園までを毎分60mの速さで歩き、公園から学校までを毎分120mの速さで走って、ちょうど25分で学校についた。家から公園まで、公園から学校までにかかった時間をそれぞれ求めよ。
プレテスト問題 2	ある人がA地から峠を超えて2700m離れたB地へ行った。A地から峠までは分速50m、峠からB地までは分速70mで歩いて、全体で46分かかった。A地から峠まで、峠からB地までの道のりをそれぞれ求めよ。
プレテスト問題 3	1周4000mの湖の周回道路がある。弟は自転車で、兄はジョギングでまわることにした。弟と兄が逆の方向に出発すると10分後に出会い、同じ方向に出発すると50分後に弟は兄に1周差をつけて追いつくという。弟と兄の速さをそれぞれ求めよ。
プレテスト問題 4	濃度6%の食塩水と濃度11%の食塩水とを混ぜて濃度8%の食塩水500gを作りたい。それぞれ何gずつ加えるとよいか求めよ。食塩水の濃度の関係式は次のとおりである。 $\text{食塩水の濃度 (\%)} = \frac{\text{食塩水内の食塩の量}}{\text{食塩水の量}} \times 100$

被験者に操作させた。その後システムを使用して距離に関する問題を被験者だけで1問教えさせることで学習を行った。ただし、システム使用時間が60分を超えた場合、また、被験者が最後の答えが出るまで教えることを諦めた場合、そこで実験を終了した。

システム使用后、問題を解くことができた被験者に対してLBTが行えていたかどうかを評価するアンケートを行った。アンケート項目を以下に示す。

問1：仮想生徒に教えている感覚はどのぐらいありましたか

(0：なかった, 1：少しあった, 2：あった, 3：かなりあった)

問2：あなたが言った内容が仮想生徒に伝わったとどれぐらい感じましたか

(0：感じなかった, 1：少し感じた, 2：感じた, 3：かなり感じた)

問3：これで仮想生徒が教えたような問題を1人で解けるようになったと思いますか

(0：思わない, 1：少し思う, 2：思う, 3：かなり思う)

アンケート後、システム使用前に行ったプレテストのうち、具体的な数値だけを変更したポストテストを行い、学習によってどの程度学習内容の理解が向上しているかを比較した。

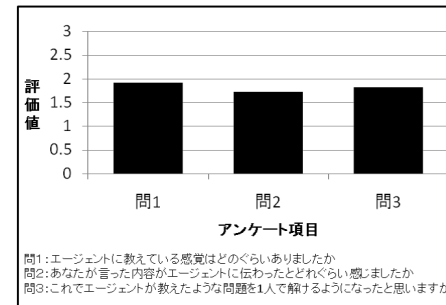


図 8 アンケートの結果  
Fig. 8 Result of Questionnaire.

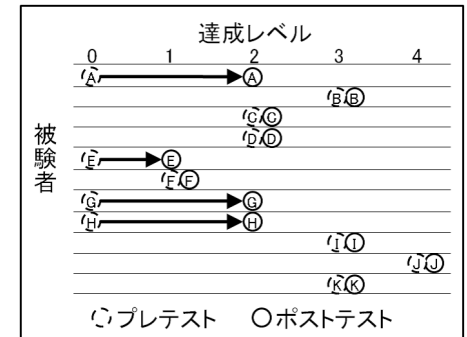


図 9 プレ・ポストテストの結果  
Fig. 9 Result of Pre-Test and Post-Test.

最後に、システム使用時の被験者の思考やテストの解答の出し方について調べるために、インタビューを行った。

また、システム操作のログとして、マウスカーソルの状態(停止・移動中)とその時間、入力した文章と入力にかかった時間、押したボタンの種類とその時間を取得し、システム操作中の様子をビデオ撮影した。

#### 4.3 結果

システムを使用して問題を解くことができた被験者数は24人中11人であった。問題を解くことができた被験者のアンケート結果を図8に、プレテスト・ポストテストの結果を図9に示す。

#### 5. 考察

##### ●教えている感について

問1, 問2, 問3の結果から本システムは教えている感覚をユーザに与えることができた。これは、仮想生徒と会話するというシステム設計をユーザが理解し、ユーザがあたかも仮想生徒に教えている感覚を与えることができたことを示す。このことから本システムはLBTシステムとして運用可能であると考えられる。

##### ●学習効果への影響

プレテスト・ポストテストの点数の結果から、学習を完了した被験者のうち、特にプレテストで全く解答ができなかった被験者の点数に増加がみられた。問題ごとに見ると、システ

ムを使い仮想生徒に解かせた問題と数値を変えただけの問題 1 は、被験者全員が解けていた。これは、システムを使って解いた問題について、問題の解法を学習できたためと考えられる。

また、問題 1 と使う知識は同じだが求める解の種類が違う問題 2 を解けるようになった被験者が 11 人中 3 人にみられた。これは、システムを使うことで問題の具体化を被験者が意識するようになり、具体化することで問題に合わせて知識を適用できるようになったためと考えられる。これは、システムを使って解いた問題だけでなく、問題に合わせて知識の適用方法について、システムを使うことで学習できている可能性を示唆しており、提案システムにより、単純な解き方の暗記ではなく、問題の構造とその分割統合の方法まで被験者が学習できることを示していると考えられる。

#### ●システムを使って問題を解くことができなかった原因

解くことができなかった被験者が 13 人いた原因を考察する。この 13 人が入力した文章のログを分析した結果、解くことができなかった被験者 13 人中 8 人が初めて (iii) 問題の具体化を仮想生徒に教えるところで諦めていたことがわかった。この原因について分析したところ、以下の 2 点の状況が見られた。

##### ● 入力の省略

ログから「家+公園+学校までの距離=家から学校までの距離」や「家から公園までの時間+公園から学校までの時間=学校までの時間」という入力が見られた。これらは、「〇〇から△△までの距離」と入力するのを手間に感じたために、意味が通ると被験者が考えた範囲で一部を省略して入力したと考えられる。本システムでは (iii) の解決における部分問題を導出するための解答は完全一致しなければならず、ここから先に進めなくなっていた。

##### ● ユーザが言葉を具体化できない

ログから「距離=家から公園までの距離+公園から学校までの距離」という入力が見られた。これは、「距離」を問題に合わせて具体化することができていないため発生している。

入力に手間がかかることを解決する方法として、システムの使用データに省略した答えも含める方法が考えられる。しかし、これは省略した答えを全て網羅しようとするデータ作成に多大な労力がかかると考えられるため現実的ではない。

ユーザが言葉を具体化できないことも含め、これらを解決する方法として、文章を作る語群の粒度を変化させる方法が考えられる。現在の実装のように粒度が細かい場合 (例: 家, 公園, 学校, から, まで, の, 距離), 語群から語を選択して入力する回数が多くなる。そのため、正解となりそうな文章が多量に作成でき、そのうちの短いものである省略の多い文

章をユーザが作る可能性が高くなると考えられる。一方、粒度が大きい場合 (例: 家から, 公園まで, 公園から, 学校まで, の距離), 文章を省略しようとする文が構成できない。また、例えば「距離」に連体修飾を示す助詞の「の」を付けると、そこで具体化の必要があることが直感的に理解できる。しかし、粒度が大きい場合、ユーザが自身の力で考えて文章を作る点に悪影響が懸念され、学習効果が発現しない可能性がある。そこで、最適な語群の粒度の大きさを検討する必要がある。

## 6. ま と め

本研究では、LBT を用いて、学習者に段階的に階層構造を持つ問題を解かせることで、知識のみならずその適用場面を学習させる個人学習支援システムを提案、実装、その評価を行った。

プレテスト・ポストテストによる客観的評価およびアンケートによる評価の結果、提案システムは教えている感覚を与えることができ、提案システムにより解いた問題だけでなく提案システムで解いていない形式の問題に対しても、それに合わせて知識を適用できることがわかった。このことから、提案システムにより、問題に合わせて知識の適用場面・方法について学習できる可能性が示された。

しかし、被験者の半数以上の 13 人が、システムを使用して問題を解くことができなかった。この原因は、ユーザが文章を入力することを手間だと感じたという、インタフェースの問題と、ユーザが言葉を具体化できないという問題がある。この問題の改善のために、文章を作る言葉の最適な粒度を調査する必要がある。

さらに今後の課題として、学習者がシステムを使って問題を解くことができないという状況を少なくするために、学習者に情報をタイミングよく与えることや、学習効果を下げずに学習者に与える情報を増やすことについて調査する必要があると考えられる。

## 参 考 文 献

- 1) Frager, S. and Stern, C.: Learning by teaching, *The Reading Teacher*, Vol.23, No.5, pp.403-405 (1970).
- 2) K, L. and G, B.: Designing Learning by Teaching Agents: The Betty's Brain System, *International Journal of Artificial Intelligence in Education*, Vol.18, No.3, pp.181-208 (2008).