DOCUMENT RESUME

ED 454 836 IR 020 815

TITLE	ICCE/ICCAI 2000 Full & Short Papers (Student Modeling).
PUB DATE	2000-11-00
NOTE	75p.; In: Learning Societies in the New Millennium: Creativity, Caring & Commitments. International Conference on Computers in Education/International Conference on Computer-Assisted Instruction (Taipei, Taiwan, November 21-24, 2000); see IR 020 792. Some figures contain very small and illegible type.
PUB TYPE	Collected Works - General (020) -- Speeches/Meeting Papers (150)
EDRS PRICE	MF01/PC03 Plus Postage.
DESCRIPTORS	*Computer Uses in Education; *Educational Technology; Elementary Secondary Education; Foreign Countries; Higher Education; *Instructional Design; *Models; *Tutoring
IDENTIFIERS	*Learning Environments; Mental Models

ABSTRACT

This document contains the following full and short papers on student modeling from ICCE/ICCAI 2000 (International Conference on Computers in Education/International Conference on Computer-Assisted Instruction): (1) "A Computational Model for Learner's Motivation States in Individualized Tutoring System" (Behrouz H. Far and Anete H. Hashimoto); (2) "A Fuzzy-Based Assessment for Perl Tutoring System" (Tang Ya, Keith C. C. Chan, Albert Wu, and Pinata Winoto); (3) "An XML-Based Tool for Building and Using Conceptual Maps in Education and Training Environments" (Juan-Diego Zapata-Rivera, Jim E. Greer and John Cooke); (4) "Controlling Problem Progression in Adaptive Testing" (Roger E. Cooley and Sophiana Chua Abdullah); (5) "Development and Evaluation of a Mental Model Forming Support ITS--The Qualitative Diagnosis Simulator for the SCS Operation Activity" (Toru Miwata, Tatsunori Matsui, Toshio Okamoto, and Alexandra Cristea); (6) "Intelligent Interactive Learning Environment: Design Issues" (Siu-Cheung Kong and Lam-For Kwok); (7) "Microgenetic Analysis of Conceptual Change in Learning Basic Mechanics" (Gary Chon-Wen Shyi and Shih-Tseng Tina Huang); (8) "Peer Help for Problem-Based Learning" (Susan Bull and Jim Greer); (9) "The Research on Difficulty of Asynchronous Learning Materials Based on Studying Time Distribution" (Wu-Yuin Hwang and Rueng-Lueng Shiu); and (10) "Using Decision Networks for Adaptive Tutoring" (Peng-Kiat Pek and Kim-Leng Poh). (MES)

ICCE/ICCAI 2000 Full & Short Papers (Student Modeling)

Proceedings

Content

▶Full & Short Papers (Student Modeling)

▲ HOME

A Computational Model for Learner's Motivation States in Individualized Tutoring System

Behrouz H. Far* and Anete H. Hashimoto**
** Faculty of Engineering, Saitama University*
*** Technical Development Department, IDO Corporation*
Shimo-okubo, Urawa 338-8570, Saitama, JAPAN
Tel: + - - - Fax: + - - -
E-mail: far@acm.org

A goal of the research is to develop an intelligent tutoring system (ITS) that adapts the delivery of instruction according to the learner's needs by taking into account learner's motivation states. Long-term and short-term parameters involved in the learning process are identified. We have found that learner's motivation has strong influence on the learning achievement. A computational model to represent learner's motivational states, using Bayesian network, is proposed. This model is further used to plan the individualized tutoring actions. This probabilistic model is the key to represent both learner 's knowledge and motivational states.

Keywords: **ITS, Student Modeling, Motivation, Bayesian Network**

1 Introduction

When designing an ITS system, usually, the first consideration is the teaching side, that is, deciding what to teach, what teaching strategies to apply, and what sequence of instruction to follow to facilitate learning. Although all these tasks are of unquestionable importance, to whom to teach, that is, the learning side should not be ignored. Teaching involves knowing what the learner wants or needs to study and planning the teaching material that leads to the desired learning outcome. However, since learners come with different background knowledge and needs, planning the individualized tutoring is not a trivial task.

Both background knowledge and motivational states of the learner have strong influence on the learning outcomes. Educational psychologists have revealed that human's motivational states are the driving forces for learning. In other words, no matter how attractive the lecture is, the learner will not benefit from it if he/she does not have the willing to engage in the learning process. But since bandwidth between the teacher and learner in a conventional classroom environment is relatively unlimited, human teachers may have a chance to bring the unmotivated learner back to the class. In the virtual classroom, the virtual tutor must be equipped with a mechanism to increase learner's knowledge via diagnosis of learner's motivational states and plan the tutoring while keeping learner motivated. We will see that, although motivation cannot be transferred from person to person, there are some principles explaining the increase (or decrease) of motivation.

A goal of this research is to develop a framework of an intelligent tutoring system (ITS) that adjusts instructions to the individual learner's needs by taking into account the motivational states of the learner [1]. A key task is increasing the bandwidth between learner and the ITS system. In order to increase the bandwidth, one must find out the hidden relationship in the learner's behavior and observed learning outcomes. Usually, learning outcome is associated to the learner's knowledge level, only. In this research, we first observe what actions contribute to increase learner's motivation to engage in the tutoring and then to plan a course of actions.

2 The Nature of Human Learning

What makes the learning process easy for one and hard for others? Looking for the answer is the primary concern of educational psychologists. In this section, we look further into the parameters influencing the human learning process.

2.1 Human learning parameters

In educational psychology individual's learning aptitude difference is explained in terms of several external and internal causes [2-5]. The external sources are usually associated to causes beyond the learner's control like the type of media or the learning environment that affect the quality of learning outcome. On the other hand, internal sources are associated with the learner's own parameters like abilities and motivations. In this work, we focus on the internal causes.

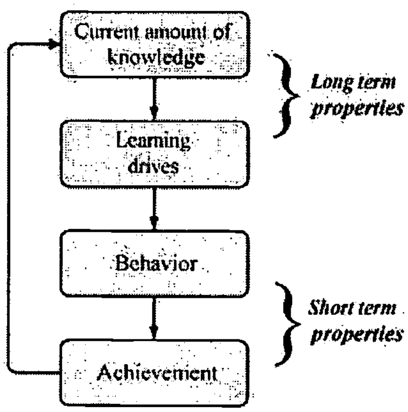


Figure 1 Human learning parameters

Figure 1, shows the set of parameters that influence the learning process. Each block in the model represents the set of parameters that describe the learning process, and the arrows indicate the direction of the influence. The first two parameters depict the learner's intrinsic characteristics. It comprises the learner's current amount of knowledge and aspects describing the learner's unconscious learning drives, like the motivation to learn. The learner's characteristics, in turn, are relevant to his/her behavior. It comprises the conscious learning drives used to measure how much effort his is putting to learn the new material. As the result of the learner's behavior, his/her achievement can be measured by the learning outcomes. The achieved learning outcome, in turn, is fed back to the current amount of knowledge. In Section 3 we specify parameters compris ing each block.

Among the subject's characteristics, one parameter that receives special attention ineducational psychology is the motivation that drives learning. Motivation can be classified in two types: extrinsic and intrinsic [4]. Extrinsic motivators comprise external driving forces like studying to pass an exam or to receive a reward. The intrinsic motivators, on the other hand, are internal forces inherent to the individual like the interest in the subject matter or the desire to be successful. The ideal is that both motivators influence learners, but the reality is different. Usually extrinsic motivators, like grades and prizes, become the objective in the classroom. Unfortunately, extrinsic motivators tend to have a short-term effect and affect the learning activity [5,4]. The intrinsic motivators are the parameters that generate learning results in long-term perspectives. The favor to intrinsic motivators can be observed in a study conducted in [6]. The explanation found is consistent with what is known about the relationship between extrinsic motivations (such as grades) and intrinsic motivation (such as challenging tasks): extrinsic motivators tend to inhibit intrinsic motivators. That is, if learners were given the choice, they would rather choose easier exams in order to get high grades than selecting more challenging tasks.

Based on this argument, the proposed tutoring system emphasizes learner's intrinsic characteristics like abilities, progress, and confidence. It does not mean that extrinsic motivators are useless (test grades are not excluded in our system). Rather, the ideal is to balance both kinds of motivating drives. In the next subsection, the theories and principles that support our idea are explained.

2.2 The motivational and learning principles

We think that learning occurs only if the learner is motivated to learn. This desire to learn, whether intrinsic or extrinsic, is the driving force of how much effort the learner is willing to put in order to learn (see Figure 1). These efforts will be measured taking into account the learner's observable behaviors such as the time spent to read a lesson or the frequency of visiting the same lesson to study. Herewith, we define the intrinsic characteristics that later will serve as the backbone of the student model.

1. ***Motivation:*** Motivational state is the force that drives the learner to engage in an activity because of a feeling of need or desire. Though motivation cannot be transferred, it may increase (or decrease)

depending on the situation that the learner is faced. One of the situations in which changes in motivational states may be observed is when the learner is presented tasks that fall in a range of challenge such that success is perceived but not certain [7]. Besides the perceived probability of success, others works [2,3] suggest also that the value of obtaining goal and acknowledge of progress are factors affecting motivation.

2. ***Learning:*** Learning is the ultimately desired change in behavior and knowledge to be achieved by the learner. Because of different background, motivational states and goals, learning results in different acquisition rate and outcomes. With regard to the factors influencing learning, readiness to understand the instruction is an essential requirement. Prerequisite knowledge is suggested as a measure of readiness. Anxiety and uncertainty of achieving goal have negative influences on learning.
3. ***Interest and progress:*** The acquisition of an ability or skill is a potential activator of interest since people tend to repeat things in which they are successful [4]. That is, when learners obtain evidence of their learning progress, not only interest tends to increase but also performance will be superior to what it would have been without such acknowledgment. Progress, may be thought of as the sum of learning achievements.
4. ***Retention:*** Retention is a measure of how well learners remember already acquired facts. The longer the time delay, the lower the retention factor. While time delay decreases retention, rehearsal strengths the ability to recall old information.
5. ***Ability degree:*** The learner's ability degree is a measure of preparedness to learn academic material [3]. We define it as directly dependent on readiness, expertise level, and complexity of the topic. Expertise level, in turn, is measured by the amount of knowledge the learner has accumulated.
6. ***Attention:*** By attention we mean a measure of how the learner is directing his/her mind to the given task. We define it as the result of the positive influence of motivation and ability degree and the negative influence of distraction due to complexity of topic.
7. ***Effort:*** The effort tells us how the learner is behaving in order to achieve learning goals. Since it is not possible to observe it directly, we measure it by the frequency of dedicating to the study (frequency of use), the time delay between studies, the amount of time engaged in reading (time for reading), whether the learner performs the tasks (practice), and whether non-mastered topics are rehearsed (rehearsal).

It is obvious that intrinsic motivators are difficult to measure. Choosing challenging tasks neither brings immediate results nor it is easily measured. Marks and points, on the other hand, are concrete measures, easily interpreted and cause immediate satisfaction. The first task is to use intrinsic motivators in the student model such that they bring immediate and measurable results. The model presented in the next section covers this.

3 Student Modeling Task

In this section, we present the student model, using Bayesian network, based on the parameters mentioned in Section 2. The student model is divided in two parts: the motivational model and the knowledge model. The motivational model is generic, domain independent and applies to all learners. The knowledge model, on the other hand, is domain specific. The subject matter chosen for knowledge model is the concepts of the C programming language.

3.1 Modeling learner's motivational states

Tutoring based on the learner's motivation requires a mechanism to diagnose motivational states. Here, we take an approach that complements the limitation of existing proposals, such as [8]. However, it may introduce a new burden in creating motivational diagnosis. It is due to the modeling process and the task of estimating the probabilities for all variables in the network. On the other hand, the advantage is that it eliminates the learner's burden because the diagnosis is running in background mode while the learner is using the system.

Building a student model based on Bayesian network requires two distinct tasks: the qualitative part that concerns the modeling of relevant variables involved in the domain, and the quantitative part that deals with the probabilities. As we are interested in representing the student motivational model, the qualitative modeling is concerned with the problems of identifying what information about the learner will be modeled and how that information will be modeled. In the quantitative modeling, we are concerned with the problem of specifying how the probabilities will be computed.

3.1.1 Qualitative analysis: encoding of dependence

The difficult part in the qualitative analysis is to find out how the variables influence each other. Our starting point was the learning parameters described in Figure 1: knowledge states, learning drives, learner's behavior, and learning achievements. These rough sets were further expanded based on the learning and motivational principles explained in Section 2. The refinement is done top-down: start from the first parameter down to the last one. The result is depicted in a network of Figure 2. The nodes in the network are divided into two types: directly observable nodes denoted by dashed-lines, and unobservable ones represented by solid-lines. The graph encodes the causal dependency among the motivational aspects relevant in the process. The common positioning of the variables is from cause to effect. An arrow from A to B is read as "A influences (or affects) B". For example, readiness is a factor that influences (or affects) motivation; ability influences attention.

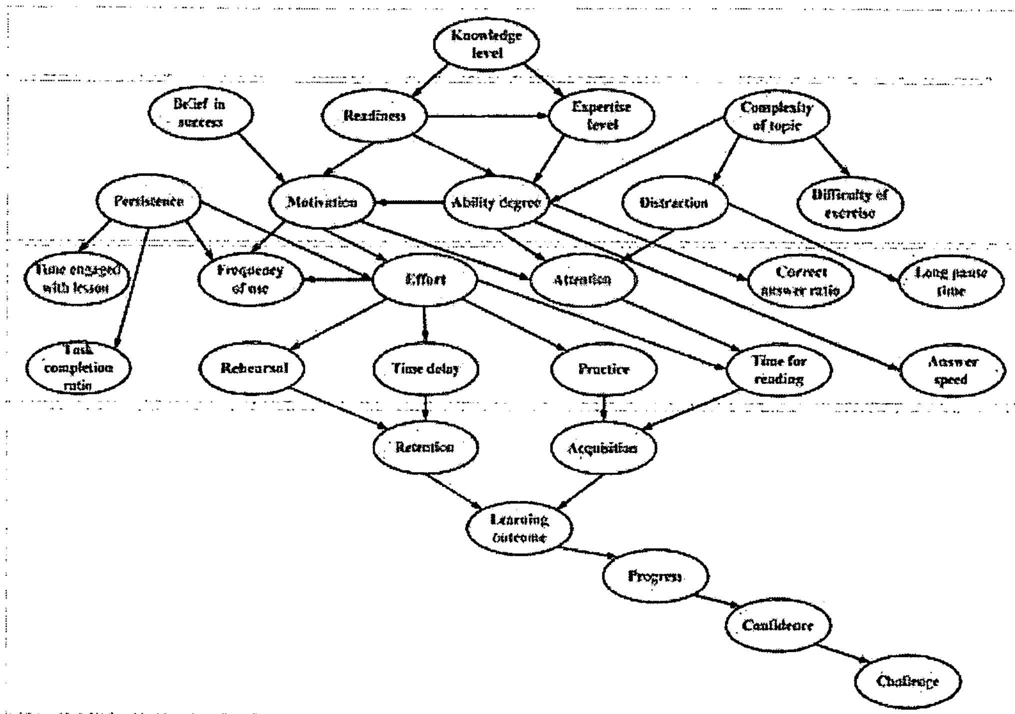


Figure 2 Student's motivational states model

3.1.2 Quantitative analysis: expressing in numbers

The nodes probabilities may come from two different sources: probabilities set by experts and probabilities coming from repetitive calibration. It is worth mentioning that obtaining exact numbers is not really crucial since we are interested in the changes between the parameters rather than the values. In many cases, the advantages of Bayesian networks outweigh the load of eliciting the numbers. For example, locally encoding of information is an important aspect. Deleting or adding new information does not require the whole network be revised.

Initially, the probabilities in the student motivational model are rough estimations. The principles behind learning and motivation were translated to sentences like:

There is a high probability that motivated learners (motivation) works harder (effort)

or

There is a low probability that the learner is persistent (persistence) if the task completion ratio is low (task completion ratio).

We repeated this example for all variables in the network. Next, the qualitative terms like high and low are expressed in numbers. Finally, using a Bayesian network editor, that we have built, those values are tested with repetitive calibration.

3.2 Modeling learner's knowledge states

Now, the qualitative and quantitative analyses for the student knowledge model are discussed.

3.2.1 Qualitative modeling: semantic of the network

The network depicted in Figure 3 represents the Bayesian network for the student knowledge models. Again there are two kinds of nodes: knowledge units and test nodes. Knowledge units represent relevant concepts comprising the domain to be taught. Test nodes represent problems that serve to verify the understanding level of each knowledge unit.

In order to build the Bayesian network of Figure 3 we start by eliciting the knowledge units comprising the domain, represented by a solid node, and ranking them according to the difficulty/complexity of the unit. For example, if a unit does not require mastery of other units, then it is a candidate to be in the easiest level. Another unit that requires just mastery of the easiest level unit is the candidate to be the second easiest level, and so on.

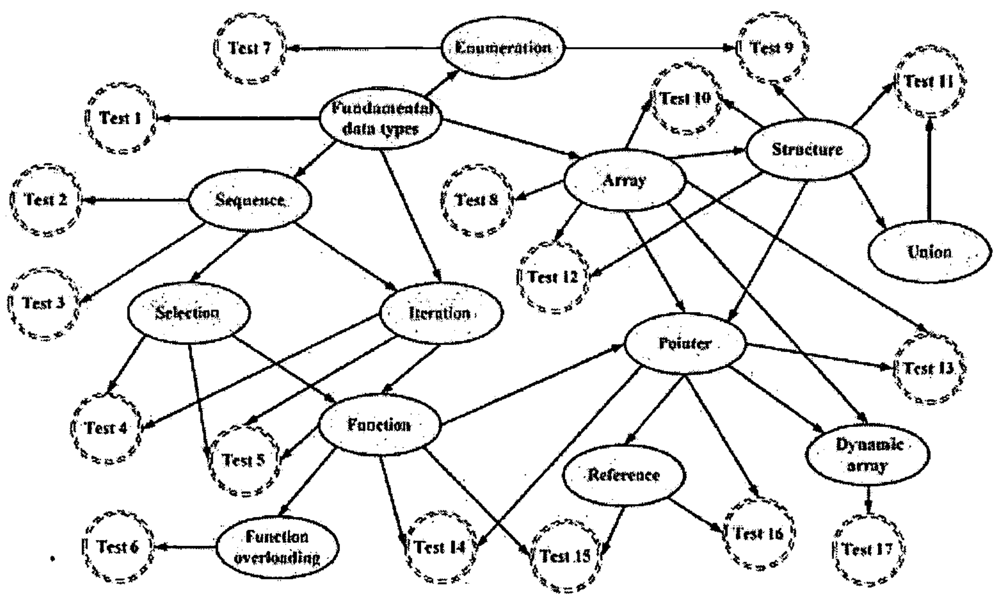


Figure 3 Student's domain knowledge model

Besides this classification, we have to find out how to represent those knowledge units in the network. Usually, Bayesian network is modeled based on cause-effect relationship. Since this is not easily perceived in our case, we extracted the factors that describe the units such as description, usage, and limitation. This analysis helps us to understand the hidden relationship between apparently unrelated units. We observed that some units fulfill the limitation of other units: for example, array and structure. In other cases, units present similarity in usage: for example, pointers and references. A link is added between those units in order to depict the fact that knowing one unit makes the probability of understanding the related unit more likely. Depending on the relevance of the knowledge unit within the domain, we can add more test nodes to the unit. In this example, since "Function", "Array", and "Pointer" play an important role within the domain of programming language, we can elaborate several test nodes covering those concepts.

The line of reasoning is as follows: if the learner solves correctly a problem associated to a knowledge unit, then the probability of knowing that unit increases. A link is added between knowledge unit and test node if it is required to know the unit in order to solve the problem. We add a link between knowledge units if exists a relevance relationship between them. Of course there is a tradeoff between compactness and preciseness. For example, learning about the "Fundamental data types" is essential for all remaining knowledge units, which we would have to add a link between that node and all other units. But, considering the precedence condition of the concepts, we were able to limit the links only to the directly relevant knowledge, such as

"Sequence expressions" and "Enumeration".

3.2.2 Quantitative modeling: dealing with probabilities

For each variable in the network, there is a conditional probability table (CPT) with respect to its parent nodes. For example, for the node "Test14", we have a CPT associating the "Test14" node to its parent nodes "Pointer" and "Function" knowledge units. That is, in order to answer the test correctly, the learner must understand both pointer and function. If the learner answers it correctly, it is inferred that he understands both units. If, however, the test was answered incorrectly, then, in the absence of other evidences, the associated units are considered not mastered yet. Suppose that we have already collected evidences that the learner knows about functions. In this case, rather than inferring both units as not mastered, it is more likely that only pointers have not been mastered yet. After including all the evidences and propagating the probabilities through the adjacent nodes, the network reaches an equilibrium state and we obtain the probability of the learner being in mastered level in each knowledge unit.

3.3 How the model works

Since learning occurs only if the learner has the desire or motivation to learn, the task we are concerned with is to keep the learner motivated to complete the tutoring. Consequently, the problems are: how to assess learner's motivational states and how to proceed tutoring in order to keep (or increase) motivation.

Let's consider the following situations: a novice learner who spends a long time without accessing the tutorial comes back to continue the lessons. Because of the long time delay between lessons, it is likely that he/she forgot something about the past lessons and needs a review. But, at the same time, the novice learner would probably become more motivated if he/she made some progress. In another case, an intermediate learner is apparently loosing motivation because of repetitive unsuccessful response to exercises.

In each case, the system can infer different treatments for each learner needs and set appropriate courses of actions. Therefore, the model will be used to perform the following tasks:

1. ***Monitoring:*** observe the learner in a sequence of interactions to adjust prior beliefs about learner's knowledge and learning drives.
2. ***Inference:*** because only a limited number of events are observable, infer what these directly observable actions tell about the other parameters.
3. ***Prediction:*** predict learner's knowledge and motivational states in the next interaction given the information currently available.

To depict the evolution of the tutoring, we represent the learning cycle as a dynamic process, as shown in Figure 4. At each interaction, the learning achievement increases (or decreases) the amount of knowledge the learner possesses in the next interaction, which indirectly increases the motivational states. Including temporal characteristic is important because if episodic interactions were considered, the learner's motivation, for example, would be inferred based on the current situation without taking into account past failure or success in outcome.

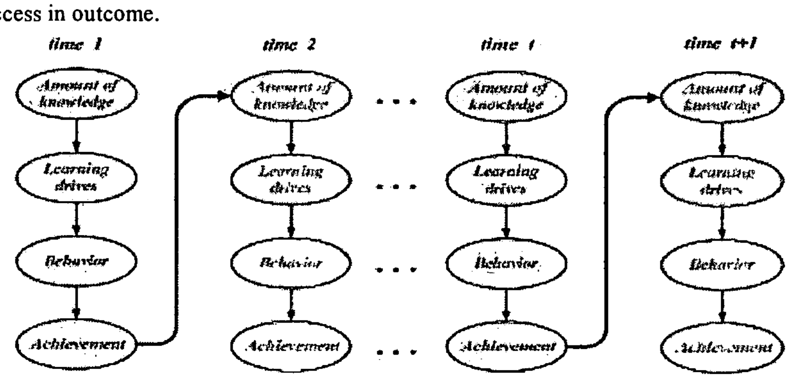


Figure 4 The dynamic process of tutoring

Dynamic Bayesian network [9] provides a mechanism to foresee the probability of interest in the next state with regard to the current beliefs. That mechanism is called probabilistic projection and can be performed by a three step updating cycle called *roll-up*, *estimation*, and *prediction* phases, as suggested in the dynamic model of Figure 4. Keeping at the most two time slices are sufficient to perform the inferential cycle. Figure 5 depicts the steps for updating a dynamic Bayesian network and below, a brief description of each step.

1. ***Prediction:*** suppose the network in Figure 5(a). Assuming that all the values have been calculated in time slice *t-1*, i.e., $Bel(X_{t-1})$, this probability should be incorporated in the next time slice by estimating $\hat{Bel}(X_t)$. In this step, the predicted probability distribution expected given the evidences known at time slice *t-1* is calculated.

   $$\hat{Bel}(X_t) = \sum_{X_{t-1}} P(X_t \mid X_{t-1}, E_{t-1}) Bel(X_{t-1})$$

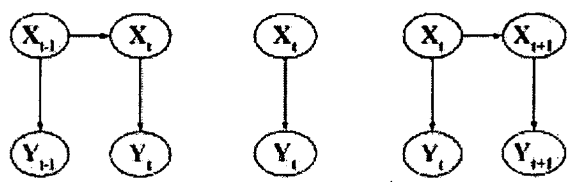
   Where E_{t-1} is all the evidence at time slice *t-1*; *P* is the probability and "^" denotes an estimation.

2. ***Roll-up:*** the roll-up is the process of removing the network on time slice *t-1* and assigning a prior probability table for the state variables at time *t*, which is the $\hat{Bel}(X_t)$ (Figure5(b)).

3. ***Estimation:*** now, using the standard probabilistic network updating, the probability distribution over the current time slice *t+1* is found and the steps for the next cycle can be repeated (Figure 5(c)).

   $$Bel(X_t) = \alpha P(E_t \mid X_t) \hat{Bel}(X_t)$$

   Where α is normalization constant.



(a) Prediction (b) Rollup (c) Estimation

Figure 5 The updating cycle

4 Planning Actions

Through educators frequently rely on experience and common sense to prepare a curriculum plan, there are some theories helping educators to organize the lessons and offer learners an easier way to assimilate new concepts. Following we describe the theories about the sequence of instruction and motivation strategies that we adopt in our project.

1. ***Theories about sequence of instruction:*** helps the instructor to select the next instruction when there are conflicting candidates.
   - *Simple-to-complex theory:* given two concepts A and B, if A is simpler than B, then choose A as the next candidate.
   - *Laws of organization:* if A and B are similar concepts and the learner knows A, then the probability of understanding B becomes higher.

2. ***Motivational Strategies:*** dictates the teaching strategies to apply given the learner's motivational states and experience. Following, some examples of strategies:
   - Whenever a less motivated or confident learner does a task well, present similar tasks that are likely to be successful in order to increase his/her confidence and motivation.
   - If the learner presents high persistence or motivation, let him/her try again the task rather than promptly presenting the correct answer.
   - Show the learner his/her motivational and knowledge states in order to stimulate self-monitoring.

The way of actually planning actions and delivering instructions will be treated in the authors' another paper.

5 Discussions

In order to model motivational states, we need a formalism that simultaneously offers mechanisms to: (a) model the causality explaining the principles involved in the learning process, (b) reason under the uncertainties inherent to the effects of the process, and (c) represent the temporal changes observed due to learning. The framework we proposed in this paper can cover all these factors. It is suitable for handling problems that can be modeled according to certain relevance conditions. In our case, the learning principles are the conditions that enable us to model the learning parameters. Although it is impossible to identify and to model all the parameters involved in learning, but Bayesian network's reasoning mechanism is capable of dealing with incomplete as well as limited amount of data. Moreover, the ability to reason about the problem without necessarily observing all the variables involved constitutes another advantage.

With respect to the computational advantages of Bayesian networks, the structure of the network allows the locally encoding of information rather than globally. That is, once the network is consistently built, each node interacts only with the directly connected nodes [9]. The gain with this property is that addition or deletion of nodes can be done locally without revising the whole network. Additionally, the computation can be performed with regard to the adjacent parameters only.

6 Conclusions

We proposed a framework for an intelligent tutoring system that adapts instruction based on the learners' needs by taking into account learner's motivation states. Our main claim is that learner's needs do not refer only to knowledge needs, but also to motivational needs. The bottleneck, however, is the limited bandwidth between human and machine. The first thought is, then, to direct the research in the latest technology in human-machine interface, like natural language understanding or eyes movement reading methods in order to increase the narrow communication channel. But is it really only the technological bottleneck that hinders the communication between human and machine? If so, then why human teachers have troubles with their pupils? This was the question that arose during the development of this work.

The bandwidth problem limits the communication channel, but providing the system all available information does not guarantee perfect communication. We realized that the cognitive and educational aspects come first. What is behind the human learning process? Why some students learn faster than the others? These questions, then, became the priorities in our work. After eliciting the parameters involved in learning, we faced with the problem of how to make best use of the limited source of information the system was capable of computing. The computational formalism that fulfilled our needs was Bayesian network. This probabilistic method not only reasons under limited source of information but also infers about yet unobserved variables. In this way, we could virtually increase the communication channel.

Planning based on motivational strategies is still in an immature stage and a subject of our forthcoming paper. For clarity, the student motivational model possesses a large number of parameters, which can be omitted according to the intended use. Since the model is modularized and domain-independent, it is also possible to reuse it to teach different domain application. The set of rules to execute motivational strategies we have defined is simply an adaptation of the motivational principles. Improvements are still needed in this direction.

References

[1] Hashimoto, A.H., Hajji, H., and Far, B.H., "A Probabilistic Approach for Student Modeling Task," 13th Annual Conference of Japanese Society for Artificial Intelligence, JSAI' 99, pp. 68-69, (1999).

[2] Eftekhar, N., Strong, D.R., and Hawaleshka, O., "Towards Dynamic Modeling of a Teaching-Learning System: The Unified Theory," International Journal of Engineering Education, vol. 12, no. 6, (1997). Also available at: http://www.ijee.dit.ie/articles/999981/article.htm (accessed on August 7, 2000).

[3] McIlrath, D.A., and Huitt, W.G., "The Teaching-Learning Process: A Discussion of Models," http://www.valdosta.edu/~whuitt/psy702/files/modeltch.html (accessed on August 7, 2000).

[4] Spencer, R., Ludington, J. and Graham, S., "Motivation and Learning: A Teacher's Guide to Building Excitement for Learning and Igniting the Drive for Quality," Peak Learning Systems, (1999). Also partially available at: http://www.peaklearn.com/chap1.html (accessed on August 7, 2000).

[5] Lumsden, L.S., "Student Motivation to Learn," ERIC Digest 92, (1994).
[6] Litchfield, B.C., and Newman, E.J., "Differences In Student And Teacher Perceptions Of Motivating Factors In The Classroom Environment," National FORUM of Applied Educational Research, vol. 12E, no. 3, (1999). Also available at: http://www.nationalforum.com/16LITCHF.htm (accessed on August 7, 2000).
[7] Knight, J., "Educational Foundation," http://ag.arizona.edu/aed/aed695a/educatio.htm (accessed on August 7, 2000).
[8] de Vicent, A., and Pain, H., "Towards Motivating Instructional Systems," In G. Cumming, T. Okamoto, and L. Gomes, editors, Proceeding of the International Conference on Computers in Education (ICCE'99), vol. 1, pp. 744-751, IOS Press, (1999).
[9] Stuart, R., and Norvig, P., "Artificial Intelligence: A Modern Approach," Prentice Hall, (1995).
[10] Hirashima, T., "Student Modeling and Adaptive Interaction," Journal of Japanese Society for Artificial Intelligence, Special Issue on Student Modeling and Adaptive Interaction, vol. 14, no. 1, (1999).

A Fuzzy-Based Assessment for Perl Tutoring System [1]

Tang Ya*, Keith C.C. Chan*, Albert Wu and Pinata Winoto*****

**Department of Computing*
Hong Kong Polytechnic University, Hong Kong
Email:{ csytang, cskcchan } @comp.polyu.edu.hk
***Faculty of Education*
Chinese University of Hong Kong, Hong Kong
Email: albertwu@cuhk.edu.hk
****School of Business and Management*
Hong Kong University of Science and Technology, Hong Kong
Email: pwinoto@ust.hk

In this paper, we present a fuzzy-based assessment for Perl Tutoring system. The Perl Tutor is implemented in a multi-domain framework so that it can teach target domain knowledge by giving supporting knowledge to reinforce the learning. In order to assess supporting knowledge, an assessment is performed *before* the tutoring begins. Its main purpose is to test student's previous declarative knowledge of computer programming. At the end of it, a directed tutoring graph will be generated to optimize the tutoring process.

Keywords: **fuzzy rule, assessment, student modeling, multi-domain tutoring**

1 Introduction

There exist many works on optimized assessment process concerned with the efficiency of testing and its completeness. *Granularity, prerequisite relationships, Bayesian propagation* and *neighborhood of knowledge states* are some of the successful attempts employed to increase the efficiency of testing [2,5,6,13,17]. Yet, even though they could increase the efficiency significantly, they still have too many burdens given the large knowledge spaces. Fortunately, not all the student models need to be precise to be useful [10]. To ease the burden to student modeling, a fuzzy approach has been used and has so far worked quite well [3,10,11].

The purpose of this paper is to present the fuzzy approach in the assessment of student's knowledge in the Perl Tutoring System [16], which teach programming language (Perl) by reinforcement from other supporting languages (C++ and/or Java). For the effectiveness of reinforcement, the system should quickly evaluate the student's knowledge of supporting languages. But the assessment needs not to be in high precision. Other works related to student modeling almost put their emphasis on the adaptive assessment *during* tutoring [14,15,17]. Yet due to the nature of our Perl tutor, we apply an assessment module *before* tutoring begins and it consists of two parts: questionnaire and testing. During the questionnaire part, students are asked to self-assess their knowledge by filling out a form provided by the system. In order to evaluate their statements, a testing part is given based on those statements. At the end of the assessment, the tutor will have a general picture of students' prior knowledge of supporting languages: with which part they are familiar etc. Since the goal of the assessment is only to get a rough knowledge states for supporting purpose, it should not take too long to complete. Thus, a coarse granularity with imprecise mastery level is appropriate.

In the next part of this paper, we briefly discuss the Perl tutoring system followed by the fuzzy logic. Then we will describe the questionnaire part and the testing part and end with discussion.

[1] The work related to this paper is funded under the Hong Kong Polytechnic University research grant No. PolyU5072/98E.

2 Overview of the Perl Tutoring System

Figure 1 illustrates the directed tutoring graph in the system [16]. The three pieces of knowledge items presented to students are: data type, logical operators and control structures. In the figure,

- Each vertex represents a sub-domain;
- Each pair of the sub-domain may be connected with a unidirectional or bi-directional arc.
- Each arc represents the relationship between two sub-domains.

Moreover, each sub-domain may consist of several vertices, which are the sub-sub-knowledge items of their parent domain. For example, under ' data type', we also have ' integer', ' float', ' boolean' etc.

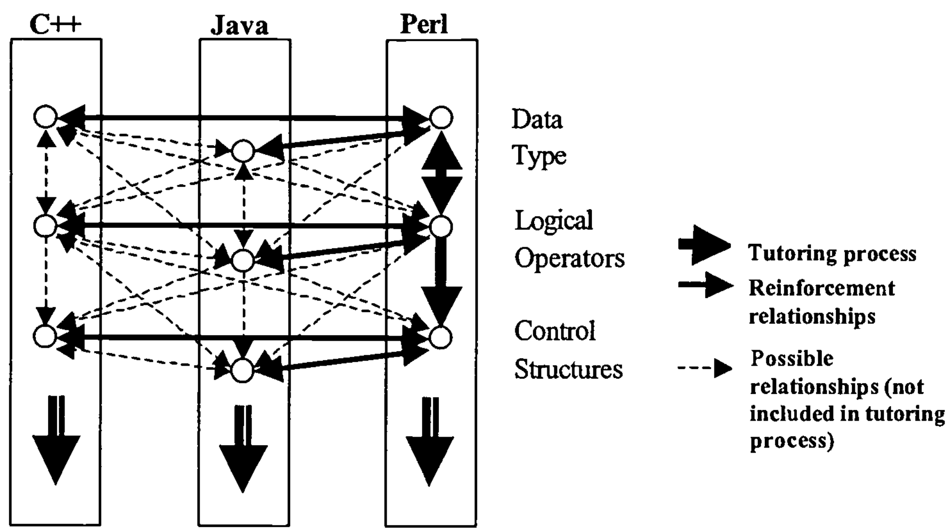


Figure 1 A Digraph of the Perl Tutoring System

C++ [1] and Java share many similarities with Perl, although they, of course, have their own features. See Table 1 for a comparison.

<table>
<tr><th colspan="3">CDR terms
(General)</th><th>Knowledge piece in PERL</th><th>Knowledge piece in C++</th><th>Knowledge piece in Java</th></tr>
<tr><td rowspan="12">Operators</td><td colspan="2">Numeric operators</td><td>+,-,*,/, %, **</td><td>+,-,*,/, %,</td><td>+,-,*,/, %,</td></tr>
<tr><td colspan="2">Relational operators</td><td><,<=,>,>=,<=>(for numeric)
lt,le,gt,ge,cmp(for string)</td><td><,<=,>,>=</td><td><,<=,>,>=</td></tr>
<tr><td colspan="2">Equality operators</td><td>=,!=(for numeric)
eq,ne(for string)</td><td>==,!=</td><td>==,!=</td></tr>
<tr><td colspan="2">Logical operators(binary)</td><td>&&,||</td><td>&&,||</td><td>&&,||</td></tr>
<tr><td colspan="2">Logical operators(unary)</td><td>!,</td><td>!,-</td><td>!,-</td></tr>
<tr><td colspan="2">Bit manipulation operators</td><td>&, |, ^, ~</td><td>&, |, ^, ~</td><td>&, |, ^, ~</td></tr>
<tr><td colspan="2">Bit shift operators</td><td><<, >></td><td><<, >></td><td><<, >>, >>></td></tr>
<tr><td colspan="2">Auto-increment & auto-decrement operators</td><td>++,--</td><td>++,--</td><td>++,--</td></tr>
<tr><td rowspan="2">Speical operators</td><td>Conditional operators</td><td>?:</td><td>?:</td><td>?:</td></tr>
<tr><td>Other operators</td><td>•,x(string operators)
,
>,\(dereference/reference operator</td><td>Sizeof
,</td><td>Instanceof</td></tr>
<tr><td colspan="2">Multiple operators</td><td></td><td>+=, -=, *=, /=, %=, &=, |=, <<=, >>=, ++=, --=</td><td>+=, -=, *=, /=, %=, &=, |=, <<=, >>=, >>>=, ++=, --=</td></tr>
<tr><td colspan="2"></td><td></td><td></td><td></td></tr>
<tr><td rowspan="2">Control structures</td><td colspan="2"></td><td>If,if/else, unless/else,
While, do/while, for, continue, goto

Notes:labled loops can be used within for, while, or do</td><td>If, if/else, while, do/while, for, continue, goto, switch, break</td><td>if, if/else, while, do/while, for, continue, break, switch

Notes:labled loops can be used within for, while, or do</td></tr>
<tr><td colspan="2">Special structures</td><td>Foreach
Redo</td><td>Exit</td><td></td></tr>
</table>

Table 1 Similarities and differences in C++, Java and Perl

CDR represents 'cross-domain reference' which serves as a dictionary for the domains. It is composed of basic terms used across the computer language regardless of which language is being referred. If the student has learned computer language before, he will develop a clear picture of the terms or concepts used, which serves as a guide for the learning of Perl. Besides, he will also integrate his former learning into his current. Through this knowledge transfers, the time spent on learning Perl will be greatly reduced [8].

Before tutoring begins, a weight is assigned to every direction of arc that represents the easiness of the acquisition of one sub-domain (target) after acquiring another (source). Since different students have different knowledge levels, the weight assigned to the same arc may not be the same. Thus, the weight across domain is jointly determined by the student model and the characteristics of knowledge (for detailed explanation, refer to [16]), i.e.,

$$w_{ij} = f(\mathbf{d}_{ij}, \mathbf{m}_{ij})$$

Where, w_{ij} is the weight of arc from i to j.

$f: R^n \times R^m \rightarrow R_+$ is a non-decreasing function.

$\mathbf{d}_{ij}$ is an n-dimensional vector representing the similarity of i and j. $\mathbf{m}_{ij}$ is an m-dimensional vector representing the student model, i.e., the student's knowledge level of i.
The dimension of $\mathbf{d}_{ij}$ and $\mathbf{m}_{ij}$ depends on the number of attributes considered. Moreover, the value of $\mathbf{d}_{ij}$ is predetermined and the value of $\mathbf{m}_{ij}$ is determined based on the student model. Thus, the system would carry an assessment module to test the knowledge of a student towards a specific supporting domain knowledge *before* tutoring begins. In this paper, we focus on the determination of $\mathbf{m}_{ij}$.

3 The Assessment Model—A Fuzzy Approach

Since the main purpose of the model is to test student's overall abilities, it is not necessary for us to gain a very accurate picture of it (although it helps). And somehow we also cannot gain a clear picture of student history. Thus, we choose a fuzzy approach in analyzing the student's performance, and we believe that the imprecise assessment of the student's prior knowledge level is adequate.

3.1 The 'neighborhood of knowledge states'

The *knowledge state* has been defined as the subset of knowledge items from a large item pool that can be mastered by students [4]. Remember that knowledge items in different domains are identified by their names, which in turn are determined by a cross-domain vocabulary. Besides, each item is characterized by its relationship with other items. The neighborhood of a knowledge state was defined by Falmagne and Doignon [7] as all other states within a distance of at most one. It has been utilized for adaptive assessment by Dowling *et. al.* [6]. In our system, we will not measure the exact distance within knowledge items, but we adopt it from another perspective. We define the neighbors of a knowledge item as the possible knowledge items which could be mastered in association with it. Let us have a look at an example.

Example 1.

1. '<','<=' represent 'less than' and 'less than or equal to' respectively, and they are relational operators.
2. '>','>=' represent 'greater than' and 'greater than or equal to' respectively.
3. '==','!=' represent 'equal to' and 'not equal to' respectively, and they are equality operators.
4. '<','<=' can be used for both numeric and strings.
5. '>','>=' can be used for both numeric and strings.
6. '==','!=' can be used for both numeric and strings.
7. Numeric is data type.
8. Strings are data type.
9. The relational and equality operators can be used for all data types, numbers, expressions or their combinations.

Let $M_S(X)$ denotes the student is sure to have mastered X. And $M_l(Y)$ denotes the student is likely to have mastered Y. Where X,Y are sets of knowledge items. Then,

$M_S(X)$□ $M_l(Y)$ can be interpreted as "if the student is sure to have mastered X, then he/she is likely to have mastered Y."

Then we will have:

1. $M_S(1)$□ $M_l\{2,4,7,8\}$
2. $M_S\{3\}$□ $M_l\{6,7,8\}$
3. $M_S\{4,5,6\}$□ $M_S\{7,8\}$
4. $M_S\{9\}$□ $M_S\{1,2,3,4,5,6,7,8\}$

For example, if the student knows well how to make comparisons for numeric and strings, then we assume that he/she is sure to have mastered: what is numeric, what is a string and the usage of the operators. Although we cannot determine that whether he masters other data types or not (that is, he is likely to have mastered other data types such as float etc), we can assess student's knowledge state without having to extensively test his abilities of each knowledge item he/she may have learned. Therefore, test items in our model may test knowledge items in a wider ranger than similar work by Collins *et. al.* [2].

3.2 Fuzzy Logic

To express precisely the notion "sure", "likely" or "unlikely", we adopt fuzzy set methods and therefore using fuzzy rule for the inferences. For example, we define

Answer = {True, False}. And A1, A2 $\subset$ Answer, thus

$$Ai = \mu_{Ai}(T)/True + \mu_{Ai}(F)/False$$

Confidence = {unlikely, likely, sure}. And B1, B2 $\subset$ Confidence, thus

$$Bi = \mu_{Bi}(u)/unlikely + \mu_{Bi}(l)/likely + \mu_{Bi}(s)/sure$$

Assume we have two rules: R1: A1 $\rightarrow$ B1 and R2: A2 $\rightarrow$ B2

Then, by Mamdani's direct methods:

$$B' = A' \circ R$$

Where, $R = R1 \cup R2$

$$Ri = \begin{pmatrix} \mu_{Ri}(T,u) & \mu_{Ri}(T,l) & \mu_{Ri}(T,s) \\ \mu_{Ri}(F,u) & \mu_{Ri}(F,l) & \mu_{Ri}(F,s) \end{pmatrix} \quad \text{and} \quad \mu_{Ri}(x,y) = \mu_{Ai}(x) \wedge \mu_{Bi}(y)$$

Note here that all operators used, such as: +, /, $\subset$, $\wedge$, $\cup$, and o, are defined in fuzzy domain.[2]

To illustrate it, let us assume that A1 is "doing well in bit shift operator", A2 is "doing bad in bit shift operator", B1 is "understand bit manipulation if doing well in bit shift operator", and B2 is "understand bit manipulation if doing bad in bit shift operator". Then, we can assign values such as:

A1 = 1.0/T A2 = 1.0/F

B1 = 0.5/l + 0.5/s B2 = 1.0/u + 0.1/l

And satisfied: R1: A1 $\rightarrow$ B1 and R2: A2 $\rightarrow$ B2. Thus,

$$R1 = \begin{array}{cc} & \begin{array}{ccc} \mu_{B1}(u) & \mu_{B1}(l) & \mu_{B1}(s) \\ 0 & 0.5 & 0.5 \end{array} \\ \begin{array}{cc} \mu_{A1}(T) & 1.0 \\ \mu_{A1}(F) & 0 \end{array} & \begin{bmatrix} 0 & 0.5 & 0.5 \\ 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$R2 = \begin{array}{cc} & \begin{array}{ccc} \mu_{B2}(u) & \mu_{B2}(l) & \mu_{B2}(s) \\ 1.0 & 0.1 & 0 \end{array} \\ \begin{array}{cc} \mu_{A2}(T) & 0 \\ \mu_{A2}(F) & 1.0 \end{array} & \begin{bmatrix} 0 & 0 & 0 \\ 1.0 & 0.1 & 0 \end{bmatrix} \end{array}$$

[2] Many books [18,19,20] in fuzzy set theory provide good explanations on these operators. We are not going to explain it further in this paper due to limited space.

With two rules, the fuzzy relation Ri is made from the implication Ai $\rightarrow$ Bi (in this case, i=1,2). The compiled fuzzy relation R is given as Mamdani' s method:

$R = R1 \cup R2$, computed as:

$$R = \begin{bmatrix} 0 & 0.5 & 0.5 \\ 1.0 & 0.1 & 0 \end{bmatrix}$$

Now, assume after a series of testing, a student performance show A' = 0.9/T + 0.2/F in doing bit shift operator. Then, we can calculate his performance in bit manipulation as:

$$B' = A' \circ R$$

$$= \begin{array}{cc} T & F \\ [0.9 & 0.2] \end{array} \circ \begin{array}{c|ccc} & u & l & s \\ \hline u & 0 & 0.5 & 0.5 \\ l & 1.0 & 0.1 & 0 \end{array}$$

$$= [(0.9 \wedge 0) \vee (0.2 \wedge 1.0),\ (0.9 \wedge 0.5) \vee (0.2 \wedge 0.1),\ (0.9 \wedge 0.5) \vee (0.2 \wedge 0)]$$

$$= [0.2 \quad 0.5 \quad 0.5]$$

$$B' = A' \circ R = 0.2/u + 0.5/l + 0.5/s$$

Which shows 0.5 likely to understand, 0.5 surely to understand and only 0.2 unlikely to understand bit manipulation.

4 Questionnaire and Testing

The questionnaire part consists of a series of knowledge items to be checked by students. The knowledge items are grouped into several groups based on the their similarities and difficulties. Then, students are asked to fill the form about their mastery level in each group. Five grades are provided for each answer, i.e., very familiar, familiar, moderately familiar, not familiar, and never heard. After students provided their answers, the system retrieves a series of testing questions based on the difficulty (upper limit) of students' answers, especially for the items marked ' moderately familiar' . But it does not mean that the presumably mastered items are not tested at all. Even the items marked ' very familiar' will be tested, but with a very low probability. Testing could be in the forms of short program lists or short questions, which are made as short, clear, and simple as possible. The reason is to avoid noise or errors which do not come from student knowledge iteself. In order to avoid ambiguity in judging knowledge level when the question is not answered well, every question only consists few higher level concepts to be handled.

Moreover, an average of membership value is used if the same item occurs in several questions. (We can use Bayesian update but with higher cost, i.e., to set all the conditional probability among every question).
For example, if from question 1, 2 and 3, a student performance on ' bit manipulation' shows

0.8/T + 0.2/F, 0.9/T + 0.3/F, and 1.0/T + 0.1/F respectively,

then the overall performance is, simply, the average, i.e., 0.9/T + 0.2/F.

If the question needed does not exist in the database, then a similar question is retrieved. The measure of similarity is based on the maximum number of high level concept appeared.

Prerequisite relationship

In addition to the neighborhood relationships, prerequisite relationships are also applied. The prerequisite relationship provides not only test item ordering criteria in a "strong" sense, but also in a "weak" sense. In ordinary prerequisite criteria P(A, B) denotes "A is prerequisite of B". In our extended criteria, we introduce A' as:

If A' is closely related to A and $M_S(A') \square M_S(A)$
then we have P' (A', B), that is, A' is weakly prerequisite of B.

So, if students have mastered item A', we have: they are sure to have mastered B without testing whether they have mastered item A or not. By doing this, we can largely tighten the testing items and thus save more time.

5 Discussion

To know student's learning history and his knowledge level, we cannot ask them too detailed questions in order to gain a more full picture of their knowledge state (although it helps) since it will make student modeling itself a kind of a complex system. But we need them to aid in the assessment, so how much trust should we have in the student's own assessment? This is the question we need answer before we proceed. In our system, we will not generate the tutoring graph solely based on their answers. Our solution is to test by giving them several pre-stored test items: if they can write out the outcome correctly, we assume that he has mastered the knowledge pieces and rules needed for this program.

Thus, the assessment will proceed. Test items need not to be like traditional testing questions in classrooms. They can be mini-programs or short questions provided that they can be used as a guide to assess students' mastery level of declarative knowledge.

Furthermore, we also should consider the nature of the language. For example, If the student has studied both Prolog and Java before, considering the respective relationship of them with Perl, we will still use Java as supporting knowledge because it is closer to Perl. This factor is called Knowledge Relation (K-R), and it will be assigned to d_{ij}.

At the end of the self-assessment section, a directed tutoring graph is generated. And student will be tutored based on it.

Currently, we are constructing the fuzzy rules which are applied for the assessment module, followed by the implementation and evaluation of it.

References

1. Capper, D.M. "Introducing C++ for Scientists, Engineers and Mathematicians", U.K., Springer-Verlag, (1994).
2. Collins, J.A., Greer, J.E. & Huang, S.X. "Adaptive assessment using granularity hierarchies and Bayesian nets", Proceedings of the 3rd International Conference on Intelligent Tutoring Systems (ITS' 96), Montreal, Canada, LNCS 1086, (1996).
3. Derry, S.J. & Hawkes, L.W. "Toward fuzzy diagnostic assessment of metacognitive knowledge and growth", Proceedings of the Annual Meeting of the American Educational Research Association, San Francisco, CA, (1992).
4. Doignon, J.-P. & Falmagne, J.-C. "Spaces for the assessment of knowledge", International Journal of Man-Machine Studies, Vol 23, pp.175-196, (1985).
5. Dowling, C.E. & Kaluscha, R. "Prerequisite relationships for the adaptive assessment of knowledge", Artificial Intelligence in Education, pp. 43-50, AACE: Charlottesville, VA, (1995)
6. Dowling, C.E., Hockemeyer,C., Ludwig, A.H. "Adpative assessment and training using the neighbourhood of knowledge state", Proceedings of the 3rd International Conference on Intelligent Tutoring Systems (ITS' 96), Montreal, Canada, LNCS 1086, (1996).
7. Falmagne,J.-C. & Doignon,J.-P. "A Markovian procedure for assessing the state of a system", Journal of Mathematical Psychology, Vol. 32, pp. 232-258, (1988).
8. Harvey, L. & Anderson, J.R.. "Transfer of declarative knowledge in complex information processing domains", http://act.psy.cmu.edu/ACT/papers/Harvey_Anderson95.html, (accessed on August 1, 2000).
9. Juliano,B.A. & Bandler,W. "Tracing Chains-of-Thought", Heidelbert: Physica-Verlag, (1996).
10. Katz, S., Lesgold, A., Eggan, G., & Gordin, M. "Approaches to student modeling in the Sherlock tutors", Proceedings of the Third International Workshop on User Modeling, Germany, pp. 205-230, (1992).

11. Lesgold, M., Eggan, G., Katz, S. & Rao, G. "Possibilities for assessment using computer-based apprenticeship environments", Cognitive Approaches to Automated, Hillsdale, Lawrence Erlbaum Associates, (1991).
12. Matsubara, Y., Nagamachi, M. "Motivation system and human model for intelligent tutoring", Proceedings of the 3rd International Conference on Intelligent Tutoring Systems (ITS' 96), Montreal, Canada. LNCS 1086, (1996).
13. McCalla, G. I., Greer, J. E. "Granularity-based reasoning and belief revision in student models", Student Modelling: The Key to Individualized Knowledge-Based Instruction, pp. 39-62. Springer-Verlag, Berlin, (1994).
14. Nguyen-Xuan, A., Joly, F., Nicaud, J.F., Gelis, J.M. "Automatic diagnosis of the student' s knowledge state in the learning of algebraic problem solving", Proceedings of the Conference on Artificial Intelligence in Education, Japan, (1993).
15. Sime, J.-A. "Modeling a learner' s multiple models with Bayesian belief networks", Proceedings of the Conference on Artificial Intelligence in Education, Edinburgh, Scotland, pp. 426-432, (1993).
16. Tang, Tiffany Ya. & Wu, Albert. "The implementation of a multi-agent intelligent tutoring system for the learning of computer programming", To appear in the 16th IFIP World Computer Congress— International Conference on Educational Uses of Communication and Information Technologies (ICEUT 2000) Beijing, China, (2000).
17. Villano, M. "Probabilistic student models: Bayesian belief networks and knowledge space theory", Proceedings of the 2nd International Conference on Intelligent Tutoring Systems (ITS' 92), Montreal, Canada, pp. 491-498, (1992).
18. Zadeh, L.A "Fuzzy sets", Information and Control, Vol. 8, pp. 338-353, (1972).
19. Zadeh, L.A. "A fuzzy-set-theoretic interpretation of linguistic hedges", Journal of Cybernetics, Vol. 2, pp. 4-34, (1973).
20. Zhu, Q. & Lee, E.S. "Fuzzy and Evidence Reasoning", Physica-Verlag, Germany, (1995).

An XML-Based Tool for Building and Using Conceptual Maps in Education and Training Environments

Juan-Diego Zapata-Rivera, Jim E. Greer, John Cooke
ARIES Laboratory, Department of Computer Science,
University of Saskatchewan, Saskatoon, Canada
Diego.Zapata@usask.ca

Conceptual maps have been used in many areas as a means of capturing and representing knowledge. Several authors have explored the use of visual tools to enhance the learning process. *Thinking maps* as well as *frame games* use visual patterns of relationships (learners thinking processes) to structure knowledge. Based on their graphical structure it is possible to recognize the thinking process(es) employed in the map. Several software applications have been created to support different kinds of maps, but they use proprietary files to represent their maps. It makes sharing of knowledge difficult and jeopardizes the widespread use of maps. This paper proposes XML (Extensible Markup Language) as the language to describe maps. A knowledge construction and navigation tool (KVT- Knowledge Visualization Tool) has been implemented using XML to represent the kinds of maps supported by *thinking maps* and/or *frame games*. This paper describes the uses of KVT in education and training environments.

Keywords: **Knowledge Construction and Navigation Systems, Conceptual Maps, Thinking maps, Frame Games, XML, and Learner Models.**

1 Introduction

Conceptual maps have been widely used in many disciplines for different purposes. Concept maps have been used in education and training as a means of capturing and representing knowledge. Concept maps are just one of a variety of visual tools employed in schools and corporations. Several authors [2,4,5,6,7,9,10, and 14] have explored the use of conceptual maps to enhance the learning process.

Several authors [1,3,7, and 14] have used map adaptation techniques in hypermedia systems to offer a pertinent group of links to a particular user in a particular situation. Existing map-based navigation systems use different adaptation techniques to change the structure of the map according to the users' goals or preferences.

In this paper, we present KTV (Knowledge Visualization Tool), a knowledge construction and navigation tool that allows students and teachers to create XML-based maps in which they can add different kinds of links to the nodes on the map and navigate throughout the content using their own map. In addition, learners can introduce their own links or use links suggested by the teacher and/or other learners. Students and teachers can remove any unwanted link and define the sequence in which the links will appear. XML-maps are viewed as an important step in the creation of an open representation of maps that facilitates sharing of knowledge and assessment of students' knowledge by comparing their maps.

2 Visual Concept-Mapping Tools

A Visual concept-mapping tools (maps) have been used for constructing knowledge and capturing information about people's thinking processes. Because of the many types of maps available, people may get confused about what kind of map to choose for a specific problem. Hyerle [4] classifies maps in three categories:

- **Informal representations**, such as *brainstorming webs, web maps*, and *mind maps*, which are used mainly to support association and creative processes.
- **Task specific maps or organizers**, such as *life cycle, text structures*, and *decision trees*, which are used in specific content areas or tasks.
- **Thinking process maps**, such as *concept maps, system thinking maps*, and *thinking maps*, which are used to represent not only content relationships on a specific area, but also the thinking process or kind of reasoning behind the map.

Web maps, *mind maps*, and *brainstorming maps* have been used to support creative processes. Their informal structure is useful in areas, such as: brainstorming sessions, decision making, problem solving, taking notes, public speaking and planning. Figure 1 shows an example of a mind map created using Mind Manager® MindJET, LLC [8].

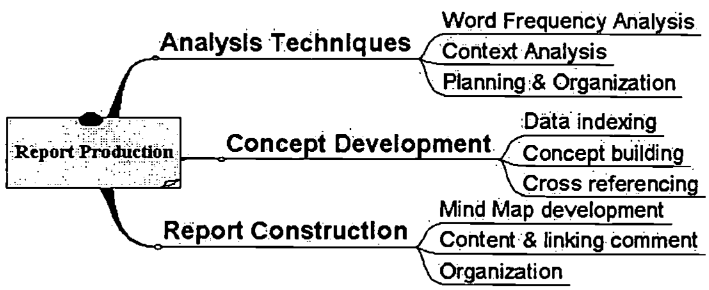


Figure 1. Example of a mind map [8].

Task-specific maps or *organizers* are designed to structure knowledge on a specific area. Figure 2 shows an example of a simple task-specific map (a classification tree) used in a biology class.

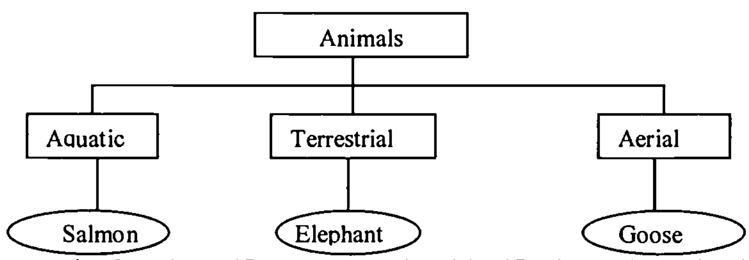


Figure 2. An example of a task-specific map or organizer (classification tree) used in a biology class.

Thinking process maps include concept maps, system maps and thinking maps. Thinking maps [4] are similar to frame games [6]. They use various kinds of visual patterns to represent information relationships and mental processes such as: sequencing, identifying attributes, cause-effect reasoning, analogical reasoning, part/whole reasoning, and classifying information.

Using *concept maps* [5,6,9, and 10] with different types of links, it is possible to represent more or less the same mental processes that thinking maps represent. The main disadvantage of *concept maps* over *thinking maps* is that their graphical structure does not necessarily reflect the thinking process. Figure 3 shows a simple example of a concept map.

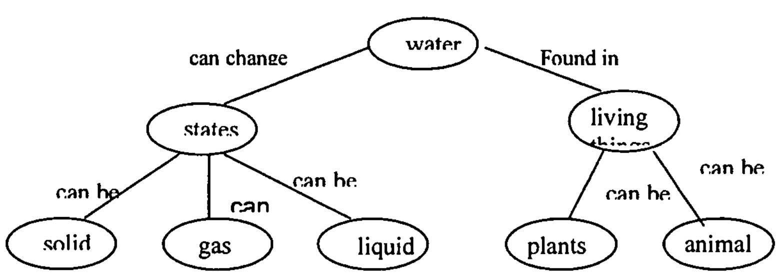


Figure 3. An excerpt of a concept map [10].

Thinking maps and *frame games* integrate knowledge views and make explicit fundamental human cognitive processes. According to Hyerle [1], by using *thinking maps*, it is possible to create any map that can be created using *brainstorming webs* and *task organizers* without being as informal as *brainstorming webs* and less content dependent than *task organizers*. Not only do *thinking maps* support structuring of content but also thinking processes, meta-cognitive abilities and reflection. Figure 4 shows some of the visual patterns supported by *thinking maps* and/or *frame games*.

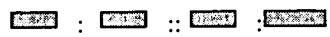
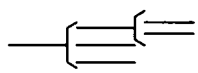
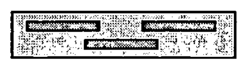
Thinking map	*Frame game*	*Reasoning*
 Bridge map	 Analogy pattern	Metaphorical
 Multi-Flow map	 Cause-Effect pattern	Systems dynamics
 Brace map	 Part/Whole pattern	Inductive and deductive

Figure 4. Some of the maps (visual patterns) supported by thinking maps and/or frame games.

3 Proprietary Map File Formats vs. XML-Maps

Most of the available commercial products (i.e. [8,11, and 13]) support *mind maps* or variations of them for multiple purposes (i.e. brainstorming sessions, decision making, problem solving, taking notes, public speaking, etc.). These products provide links to external applications, to other maps, and to content on the web. Although, ThinkingMaps® [12] is a software tool for the creation of thinking maps in education and training environments, it does not provide links to external applications, to other maps, or to the web. All these products use proprietary map file formats to represent their maps. It makes difficult sharing of knowledge and jeopardizes the general use of maps.

Using XML as the language to represent maps it is possible to eliminate proprietary files. The creation of a DTD file*(Document Type Definition) to validate XML-maps should consider the main characteristics of the maps, such as: linking nodes to external applications, to content on the web, and to other maps. The DTD file proposed in this paper ('XMLmaps.dtd') covers all of the eight kinds of maps supported by *thinking maps* [4] and the ten kinds of maps (visual patterns) supported by *frame games* [6]. We have chosen to work with *thinking maps* and/or *frame games* because of their property of providing different visual patterns to represent different thinking processes. Figure 5 shows a fragment of the DTD file created to validate XML-maps.

Some of the benefits of using XML as the language to represent *thinking maps* and/or *frame games* are:

- XML provides an open format to maintain and share maps as opposed to proprietary file formats.
- By using a common vocabulary in conjunction to XML-maps, it is possible to compare maps. That is, maps can be compared to find similarities and differences in the type of structure employed (thinking process(es) used by the learner to analyze the topic), relation among nodes and types of links and documents attached to each node.
- Any XML query language such as XML-QL or XQL can be used to create queries to compare maps. By comparing maps it is possible to assess learners' knowledge and determine possible misconceptions, or gaps on a specific concept or group of them. By analyzing the type of map used to represent the knowledge it is possible to identify possible problems of the learner with a specific kind of reasoning.
- XML permits collaborative viewing of maps. See section 4.3 (KVT- Navigation System).
- By maintaining the student's knowledge information (XML-maps) in the learner model, new interesting opportunities for assessment, collaboration, adaptation, and inspection can be explored.

- Opening visual knowledge representations is an important step towards the goal of capturing, sharing, and using knowledge across disciplines.

Figure 6 shows a fragment of an XML-map used to study Anatomy. This map has been validated using the grammar rules encoded in 'XMLmaps.dtd'. Figure 7 shows the graphical representation of the same XML-map. This map can be classified as a 'brace map' following the notation of *thinking maps* or as a 'parts-whole' pattern using the *frame games* representation. In both cases, they represent part-whole relationships among concepts and inductive/deductive kinds of reasoning.

```
# Author: Juan Diego Zapata Rivera
# Version: XMLmaps.dtd 1.0
<!--*****************************************************************
ELEMENT: XMLmap
COMMENT: thinking maps structures and frame games->
<!ELEMENT XMLmap  (CircleMap* | BubbleMap* |DBubbleMap* |
BridgeMap* |FlowMap* | BraceMap* | TreeMap* |MultiFlowMap* )
<!ATTLIST XMLmap
<!--*****************************************************************
ELEMENT: BraceMap
COMMENT: Brace map. whole/ part. Inductive and Deductive Thinking->
<!ELEMENT BraceMap (BraceNode+,Link*)
<!ATTLIST BraceMap
MapId CDATA    #REQUIRED
Type   CDATA    #REQUIRED
Name  CDATA    #REQUIRED
Description  CDATA    #REQUIRED
<!--*****************************************************************
ELEMENT: Link
COMMENT: Link description-->
<!ELEMENT Link (FromNode,ToNode,LinkMedia)
<!ATTLIST Link
LinkId CDATA    #REQUIRED
Type   CDATA    #REQUIRED
Name  CDATA    #REQUIRED >
```

Figure 5. Fragment of DTD for XML-maps

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="XMLmaps.xsl"?>
<!DOCTYPE XMLmap SYSTEM " XMLmaps.dtd">
<XMLmap>
    <BraceMap Type="'BraceMap'" Description="'Example of an XML-map'" Name="'The human body'"
    MapId="'Map001'">
        <BraceNode NodeId="02" Type="'BraceMap'" Name="'head'" XMLthinkingMapLink="''">
            <LinkMedia XMLConcept="'head'" Text="''" Application="''" URL="'1'" Sound="''"/>
            <Chlidren>05</Chlidren>
            <Chlidren>06</Chlidren>
            <Parents>01</Parents>
        </BraceNode>
        <BraceNode XMLthinkingMapLink="''" Name="'ears'" Type="'BraceMap'" NodeId="05">
            <LinkMedia XMLConcept="'ears'" Text="''" Application="''" URL="'www.body.xml'"
Sound="''"/>
            <Parents>02</Parents>
        </BraceNode>
        <BraceNode XMLthinkingMapLink="''" Name="'eyes'" Type="'BraceMap'" NodeId="06">
            <LinkMedia XMLConcept="'face'" Text="''" Application="''" URL="'www.body.xml'"
Sound="''"/>
            <Parents>02</Parents>
        </BraceNode>
        <BraceNode XMLthinkingMapLink="''" Name="'torso'" Type="'BraceMap'" NodeId="03">
            <LinkMedia XMLConcept="'trunk'" Text="''" Application="''" URL="'1'" Sound="''"/>
            <Parents>01</Parents>
```

Figure 6. Fragment of an XML-map about 'Anatomy'.

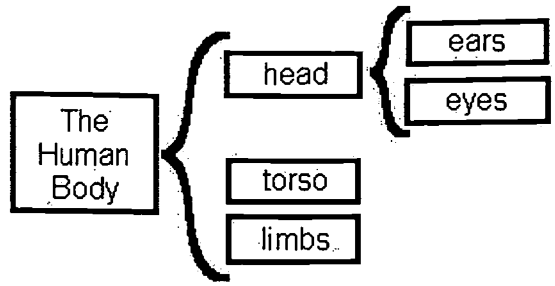


Figure 7. Graphical representation of an XML-map about 'Anatomy'.

4 KVT (Knowledge Visualization Tool)

KVT is a map construction and navigation system that allows the creation of XML-based *thinking maps* or *frame games*. KVT also provides the possibility to link different kinds of resources to specific nodes. In this way, KVT supports personalized navigation throughout the class content. Students can create their own knowledge structure using a set of predefined concepts (*common vocabulary* given by the teacher) and use their own map to access class resources. These resources are suggested by the teacher (*initial links*) or by his/her classmates during the creation of their maps (*collaborative browsing using XML-based maps*).

The class content is not limited to a specific group of pages, videos, sounds, etc. On the contrary, any student or teacher in the class can navigate through the map via the WWW, can add links, and can add new resources. Every participant has access to all of the resources that are associated with the nodes in his/her map. The list of resources attached to a node can be ordered arbitrarily by the learner.

4.1 KVT's Architecture

KVT (see figure 8) is composed of the following modules:

- ***Map Construction Tool***. KVT supports the ten kinds of maps identified in the context of *frame games* [6] and the eight types of *thinking maps* proposed by Hyerle [4,12]. Students select concepts from a predefined list and create their own structure. Having a predefined list of concepts (*common vocabulary*) makes it easier to share, compare, analyze and evaluate maps. Students can link different resources (course materials, web pages, documents stored on different applications, etc) to their map. They can even include other maps in a recursive manner. Students' maps are stored in the learner model for further modification, analysis and evaluation.
- ***The Browser***. This is the main interface to visualize one's map and its associated class content. Students can navigate throughout the content by clicking on any node of the map and selecting one of the links/documents that are available for this node. Furthermore, students can navigate freely and add links and documents to any of the nodes in the map. Students can navigate using links suggested by other students/teachers in a hyperspace created collaboratively for a particular topic and encoded on the map.
- ***The Learner Model***. The Learner module maintains basic learner information as well as their XML-maps (XML files including map structure, links, and order preferences). Students can add, order, modify, or remove links and nodes. Students and teachers contribute to populate each node with different sort of resources, but it is up to each person to remove unwanted resources and define the sequence in which he/she prefers to see the resources.
- ***Course Materials***. Class resources are classified into three main categories: web content, XML content, and general documents (text, sound, images, videos, etc.). They comprise an open range of materials that are organized first by the teacher. Using KVT, students and teachers can create different representations of the knowledge, and as a result of their contributions a highly refined subset of useful documents will be attached to each of the nodes. KVT supports the cooperative creation of information spaces to be used in educational contexts.
- ***Learners and Teachers Share Maps***. Learners and teachers can use KVT in a number of ways. For example, Teachers and learners can visualize maps using different levels of granularity. Learners can use an existing map as a guide to study the content, or use this system as a learning tool to facilitate remembering, create maps collaboratively, share their maps, and engage in interesting discussions

about a particular topic. Teachers can create maps to serve as 'guided tours', which can be used by students to navigate throughout the content. Teachers can use XML-maps to assess the student's knowledge. This can be done by comparing different maps (visually or through queries) to determine problems in the learning process for a particular student or groups of them. Finally, teachers can use this system as an adequate environment to promote reflection among students on a specific topic (map structure and content).

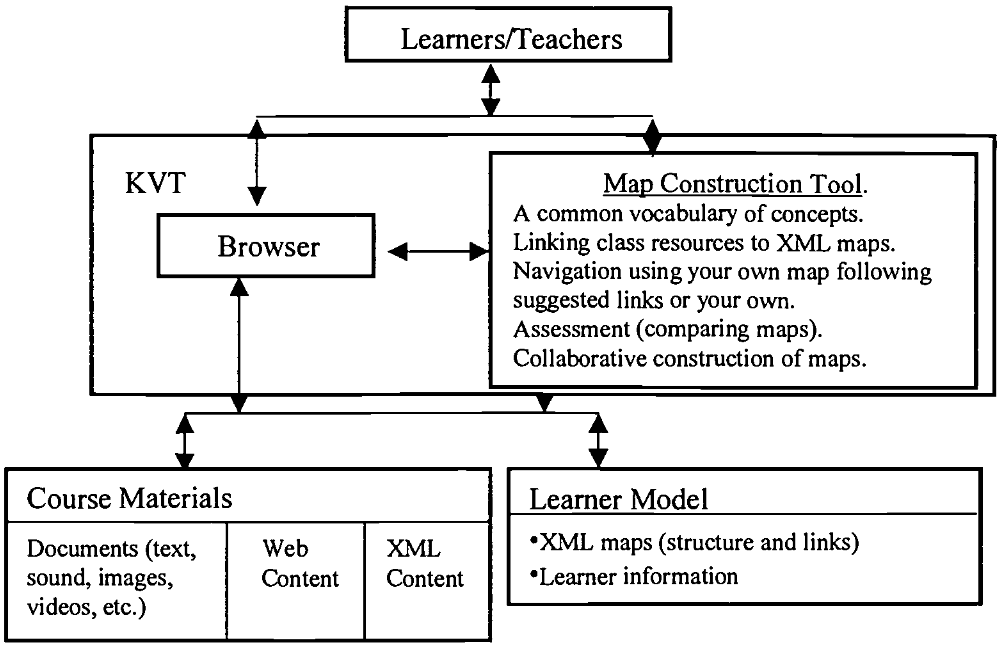


Figure 8. KVT- System Architecture.

4.2 KVT- Linking Documents to Nodes

Using KVT it is possible to add different kinds of documents to the nodes of the map. Figure 9 depicts the user interface provided by KVT to add, modify and remove documents. This interface allows students/teacher to attach a web page, XML document, video, sound, or image to any node on the map. KVT also provides the option to test any of the documents, edit the description and document fields and remove any unwanted link from the map. By clicking the headers on the grid it is possible to change the order in which the links will be presented to the student when navigating using the map. Order preferences are stored in the learner model for further use.

New links/documents for a particular node are automatically shared with all of the maps that contain such a node. This can affect maps of several students/teachers in the system. However, individual sequencing or removal of resources affects only the student's own map. Maps are stored as XML files in the learner model. The example on figure 9 shows a Brace-map that is used to organize information related to Anatomy. The grid of current documents shows the currently available links for the concept 'ears'. It is possible to visualize who included each link (user type/user), the document type location and description.

4.3 KVT- Navigation System

Figure 10 shows how students and teachers can navigate on the web using their own maps and their own links or the ones suggested by others. Just by clicking the concept, a list of current links/documents appears to be selected. If the student has not chosen any particular sequence of resource presentation, this list is initially ordered by type of user (teacher/student). In this environment, it is also possible to navigate freely on the web by entering a URL or just following the links on the current page. When an interesting page is found, it can be attached to any concept on the map by selecting the target concept and pressing the button 'Add Link' located at the bottom of the window.

The example in Figure 10 shows how the student uses the map to navigate by the links related to the concept 'ears'. The current web page corresponds to the first link suggested by the teacher 'Anatomical Tour of the Ear'.

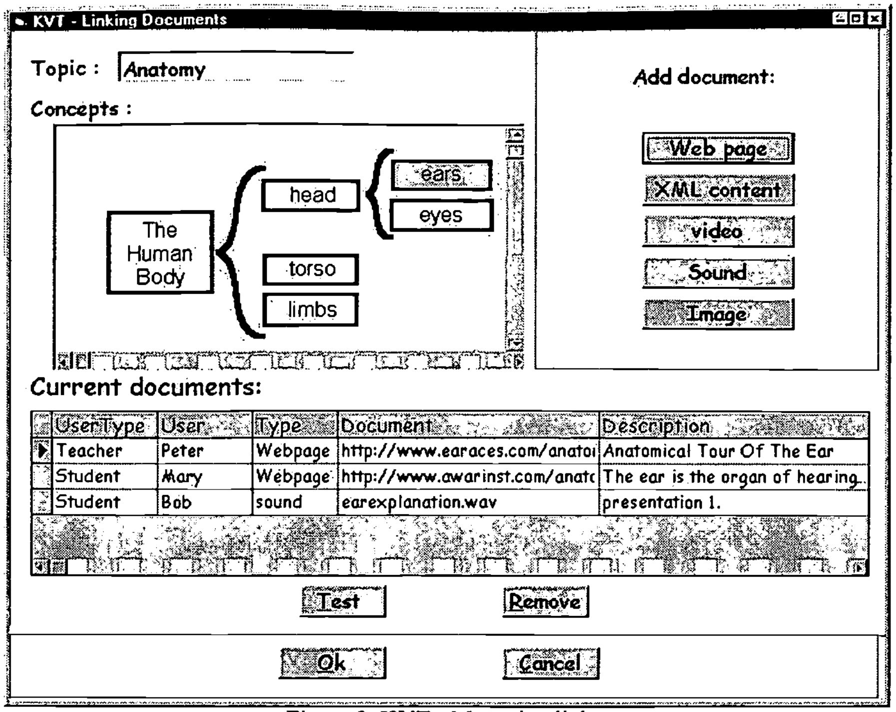


Figure 9. KVT - Managing links.

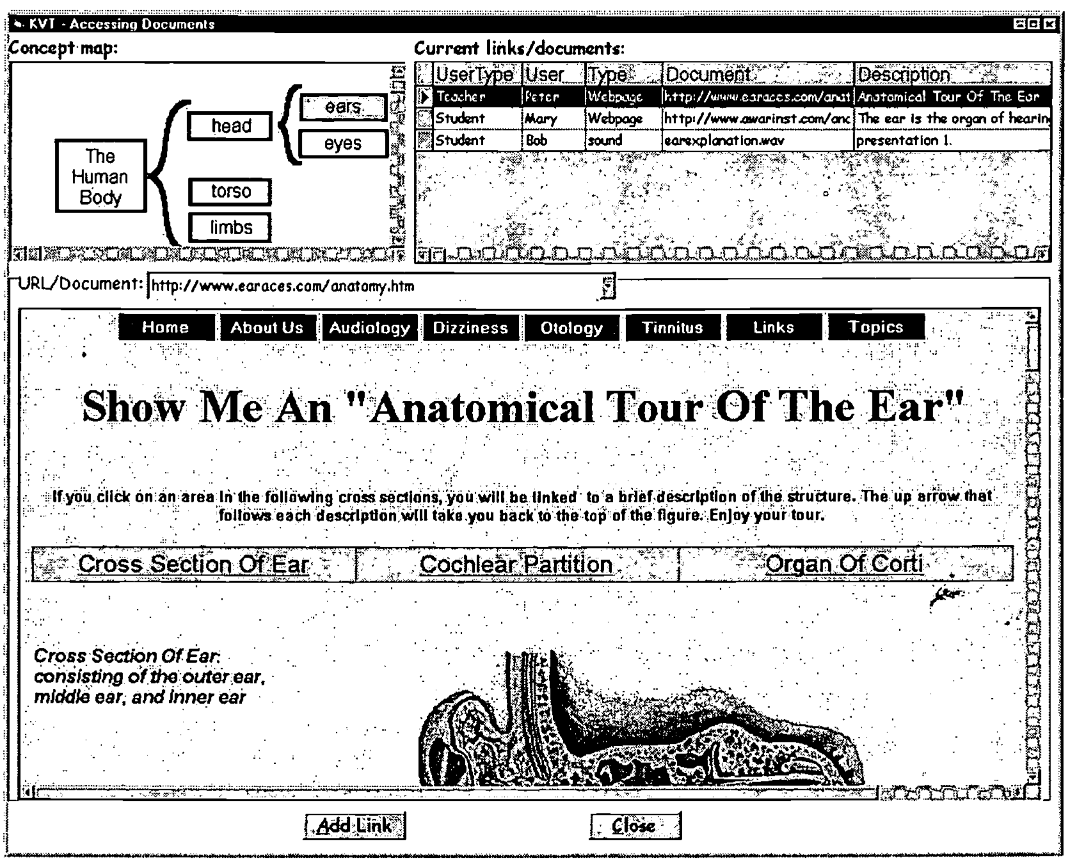


Figure 10. KVT- Navigating on the web using XML-maps and suggested links/documents.

5 Conclusions and Future Work

XML offers an excellent language to represent maps. Using XML maps, it is possible to support knowledge sharing without the problems of having proprietary files. By using a common vocabulary for the content and XML maps, it is possible to compare map structures.

XML-maps (*thinking maps, frame games*) are very useful in education and training environments because they support content structure and make explicit fundamental human cognitive processes.

KVT offers an attractive tool for the creation of maps and supports collaborative navigation throughout the content. By using XML-maps, KVT provides a better support to education or training setups that uses maps to create, share and assess knowledge. By including XML-maps into the learner model, new possibilities for visualization and inspection of XML-maps can be exploited in order to improve the learning process.

References

[1] Brusilovsky, P. "Methods and Techniques of Adaptive Hypermedia". In Adaptive Hypertext and Hypermedia. Brusilovsky, P., Kobsa, A., and Vassileva, J. (eds.) Kluwer Academic Publishers, pp. 1-41 (1998).

[2] Buzan, T. "Use both sides of Your Brain". Ed. Dutton. New York (1979).

[3] Gaines, B. R. and Shaw, M.L.G. "Web Map: Concept Mapping on the Web". (1995). Available on-line:http://ksi.cpsc.ucalgary.ca/articles/WWW/WWW4WM/WWW4WM.html

[4] Hyerle, D. "Visual Tools for constructing knowledge", ASCD press. Alexandria, Virginia, (1996).

[5] Jonassen, D., Wilson, B., and Cole. P. "Cognitive Approaches to Instructional Design". G.M. Piskurich (ed). The ASTD handbook of instructional technology. McGraw–Hill. New York, pp. 21.1-21.22. (1993). Also available on-line:http://www.cudenver.edu/~bwilson

[6] Jonassen, D., Beissner, K., and M., Yacci. "STRUCTURAL KNOWLEDGE. Technique for Representing, Conveying, and Acquiring Structural Knowledge". Lawrence Erlbaum Associates, Publishers. Hillsdale, NJ (1993).

[7] Micarelli, A., and Sciarrone, F. "A case-based toolbox for guided hypermedia navigation". In Proceedings of the Fifth International Conference on User Modeling, pp. 129-136 (1996).

[8] MindManager® by MindJET LLC. Test version V3.5.5.SP2. Apr 8, (1999).

[9] Novak., J. D.. Learning, Creating, and using Knowledge. Concept Maps as Facilitative Tools in Schools and Coorporations". Lawrence Erlbaum Associates, Publishers. Mahwah, NJ (1998).

[10] Novak., J. D., and B. D., Gowinn. "Learning how to Learn". Cambridge University Press. Cambridge, England (1984).

[11] THE BRAIN® by Natrificial Software Technologies. Free version 1.73. 9/1/99, (1999).

[12] ThinkingMaps® by Innovative Learning Group. Trial version 1.0.1, (1998).

[13] VisiMap® by CoCo Systems Ltd. Trial version 2.5 c. Nov 1999.

[14] Zyryanov, M. "Adaptive local maps in the hypermedia components of intelligent learning environments"'. In: P. Brusilovsky, P. Kommers and N. Streitz (eds.): Multimedia, Hypermedia and Virtual Reality: Models, Systems, and Applications. Lecture Notes in Computer Science, Berlin: Springer-Verlag, pp. 306-310 (1996).

Controlling Problem Progression in Adaptive Testing

Roger E. Cooley & Sophiana Chua Abdullah
University of Kent at Canterbury
Computing Laboratory, University of Kent at Canterbury,
Canterbury, Kent CT2 7NF, The United Kingdom
Tel: +44-1227-823816
Fax: +44-1227-762811
Email: rec@ukc.ac.uk, sc34@ukc.ac.uk

Adaptive testing has, in recent years, been used as a student modelling technique in intelligent tutoring systems. One of the main issues has been to optimise the progression of problems posed as the student performs the adaptive test. Previous research has concentrated on finding a structure in a fixed collection of problems. This paper describes an algorithm for problem progression in adaptive testing. After describing current approaches to the progression problem, the paper discusses the role of expert emulation. It then describes a knowledge elicitation exercise, which resulted in a solution to the progression problem. Part of the knowledge elicitation process was supported by software based on constraint logic programming, clp(FD), and the paper concludes with an assessment of the prospects of developing an extended knowledge elicitation support system.

Keywords: **intelligent tutoring system, knowledge construction and navigation, adaptive testing, constraint logic programming**

1 Introduction

The major advantages of adaptive testing over fixed item testing are that a student's knowledge is explored thoroughly and efficiently, and with a minimum of redundancy. By asking an appropriate number of problems at appropriate levels of difficulty, adaptive testing neither bores by unnecessary repetition nor intimidates by posing a series of inappropriately difficult problems [1]. This makes adaptive testing attractive for student modelling in intelligent tutoring systems [2],[3].

This research was conducted in the context of providing remedial help in mathematics to a transient population of prisoners in a local prison. Here the students are studying courses such as City and Guilds (Key Skills), City and Guilds (Number Power) and for GCSE level examinations. Working with prisoners can face tutors with problems not normally encountered in more conventional settings. Unlike school students, the prisoners not only lack uniform prior knowledge in mathematics, but tend also to join or leave the prison at individual times. This makes the job of the human tutor difficult because of the need to assess the knowledge level of each prisoner before assigning them the appropriate level of one or more of the above courses and examination. Currently, fixed item testing is used as an assessment tool. This approach has a major disadvantage. Many prisoners are 'math anxious' and the use of fixed item testing may undermine their confidence and motivation in the subject. Adaptive testing avoids this danger by presenting problems at an appropriate level of difficulty.

One of the main issues in adaptive testing is the determination of an efficient progression from one problem to another. Previous proposals have included hard-wiring prerequisite relationships between knowledge items [3], and preparing an indexing framework for problems [4]. Section 2 of this paper reviews the major lines of research; and the paper then describes an approach to the progression problem based on the knowledge acquisition techniques used for expert systems. In doing so, it continues in the vein of Khuwaja & Patel's work [5]. The paper presents a rationale for this approach, describes briefly a semi-automated method of eliciting syllabus content and characteristics, and then presents a progression technique elicited by standard techniques with an expert. It concludes with a discussion of the feasibility of automation in this area.

2 The Progression Problem

In a problem-solving environment, *problem progression* is concerned with the strategy in which the next problem is selected. In adaptive testing, this is usually based on the student's response to the current problem, as the process of selecting the next appropriate problem is crucial to the efficiency and precision of the whole student modelling process. Also, presenting *the right question at the right time* maintains the motivation of the student.

The structure of the domain, that is the way in which problems are related to one another, determines problem progression in adaptive testing; and the two significant and distinctive approaches to determining such structures are discussed in this section.

2.1 Item Response Theory

For adaptive testing systems which adopt the *Item Response Theory* or IRT [6], such as SIETTE [7] and CBAT-2 [8], the domain is made up of test items which are kept in an *item pool*. The construction of an item pool usually involves major empirical studies for content-balancing, to ensure no content area is over-tested or under-tested, and for item calibration. Each test item is associated with one or more of the following parameters – the *difficulty level, the discriminatory power and the guessing factor.* The *difficulty level* measures the difficulty level of a test item, the *discrimination power* describes how well the test item discriminates students of different proficiency, while the *guessing factor* is the probability that a student can answer the test item correctly by guessing.

Problem progression takes place like this. The adaptive test starts with an initial estimation of the student's proficiency, è. A *best* item or problem is selected. This is one which provides the most information about the student, and is calculated from the item's three parameters and current proficiency, è. An ideal item should have a difficulty level close to è, a high discriminatory power and a low guessing factor. A new proficiency, è', and its confidence level are calculated based on whether the student has answered the problem correctly or not, the old è, and the item parameters. The test continues until a stopping criterion is met, for example, when the confidence level of è' has reached a desired level.

2.2 Knowledge Space Theory

There are adaptive testing systems built on the theory of *knowledge spaces[9]*. Examples include a web-based, domain-independent system called RATH [10], a web-based system for the domain of mathematics called ALEKS [11], and a general purpose system for testing and training called ADASTRA [12].

Like the IRT-based systems, the domain is made up of test items of an academic discipline, each of which can be a problem or an equivalence class of problems that the student has to answer. The student's *knowledge state* is defined as the set of items in the domain that the student is capable of solving. For example, if a student has the knowledge state *{a,b,d}*, this means that he can solve items ***a***, ***b*** and ***d***. Not all possible subsets of the domain are feasible knowledge states. Consider the example shown in [13]. In a domain of mathematics, if a student can solve a percentage problem, (item ***d*** say), then it can be inferred that the student can perform single-digit multiplication, (item ***a*** say), and thus any state that contains item ***d*** would also contain item ***a***. The collection of all feasible knowledge states is called the *knowledge structure.* The knowledge structure must also contain the *null state {},* which corresponds to the student who cannot solve any item, and the *domain,* which corresponds to the student who can solve or master all items. When two subset of items are knowledge states in a knowledge structure, then their union is also a state. This means that the collection of states is *closed under union*. When a knowledge structure satisfy this condition, it is known as a *knowledge space*.

In practice, items for a domain are derived from instructional materials and systematic knowledge elicitation with teachers. This is also the case with establishing knowledge states where query procedures systematically elicit from human experts the prerequisite relationships between items [3], [14].

Once the domain is represented as a knowledge space, the adaptive testing strategy is then to locate as efficiently and as accurately as possible, a student's knowledge state. Problem progression becomes straightforward. For example, if a student has answered an item correctly (incorrectly), it can be inferred that he can (cannot) answer a prerequisite item and will thus not be asked to solve the latter.

2.3 Other Approaches

The domain can be represented as a *granularity hierarchy* [15] where items which represent a topic, subtopic or skill, are described at various grain sizes and connected together into a granularity hierarchy which allows focus shifts along either *aggregation* or *abstraction* dimensions. In this way, the ability to recognise student behaviour at varying grain sizes is important both for pedagogical and diagnostic reasons.

Other examples include an *indexing framework* for the adaptive arrangement of problems in the domain of mechanics [4], a problem-simplification approach [16], an optimisation expert system where both the knowledge structures of the student and the teacher are represented by structural graph, and problem progression is controlled by the relationship between the student's knowledge structure and that of the teacher's [17]. Evidence of a strong use of a student model in controlling problem progression can also be found in a system called TraumaCASE [18] which automatically generated clinical exercises of varying difficulty, and in the work of Beck, Stern & Woolf [19] who recorded information about a student using two factors – *acquisition* and *retention*. *Acquisition* records how well students learn new topics while *retention* measures how well a student remembers the material over time.

3 Knowledge Elicitation

The concern of the researchers discussed above is to exploit a structure of a syllabus to improve the efficiency of tests. The structure may either be revealed through elicitation, as was done by Dowling and her co-workers, or may be derived from a statistical analysis of student behaviour, (IRT), or it may be seen as being derived from the nature of the problem domain. Though there may be, from some given point of view, an optimal way of structuring a syllabus, the view adopted in this research is that it is a subjective matter to be determined by an expert teacher. Such a teacher might make use of informal statistical information, subject domain information as well as pedagogic information in determining a suitable structure. Studies of intelligent tutoring systems have shown that, as one would expect, it is difficult to transfer systems from one setting to another, because there is considerable cultural variation in both teaching and learning [20]. This provides the prime motive for investigating techniques based on expert emulation for the production of tests for local consumption.

Moreover, this is a natural extension of the *intelligent tutoring systems* endeavour, and it has an additional advantage. A lack of homogeneity amongst a student body can weaken the effectiveness of techniques based on population statistics; and the target body of students with which this paper has been concerned is, educationally, not very homogeneous.

4 Eliciting the Syllabus

There are several problems to be confronted when adopting an expert emulation approach to designing an adaptive test. They include the problems of finding suitable experts [21], selecting appropriate forms of knowledge representation and choosing appropriate methods of knowledge acquisition.

The approach to knowledge acquisition in the research described here is to separate the task of designing an adaptive test into the following sub-tasks:

- describing classes of problems,
- describing the skills used to solve problems,
- describing responses to problems,
- problem generation,
- problem progression based on student responses.

For the particular domain tackled, namely the arithmetic of elementary fraction addition, software has been developed to support the first four of these subtasks using Constraint Logic Programming, clp(FD), embedded in Prolog, [22]. This work has been described in a recent conference paper [23], and is briefly summarised here.

Clp(FD) is actively used by the knowledge engineer conducting knowledge acquisition interviews. The teacher, who is the target of the emulation, is not expected to write constraints, but is more than likely to take an interest in them. During discussions, which involve the production of example problems, the knowledge engineer enters the necessary constraints, or modifies existing constraints, to describe the particular class of problem under discussion. The set of constraints is then solved interactively to produce example problems. These form the basis of a discussion, and may lead to further rounds of discussion and modification.

The description of a class of problems is treated as a set of constraints. This consists of a set of variables, a statement of the domains of the variables, and a statement of the relational constraints that hold between the variables. For example, during an interview, the human tutor wanted to represent a class of problems, which involved the addition of two proper fractions with a common denominator of the form,

$$N1/D1 + N2/D2 = N/D$$

and he wanted to use single-digit integers.

This can be represented in clp(FD) as a code fragment:

```
domain([N1,D1,N2,D2],1,9),      % Single digit integers
N1 #<  D1,                      % First operand - proper fraction
N2 #<  D2,                      % Second operand - proper fraction
D1 #= D2.                       % A common denominator
```

The following is an example of the use of clp(FD) to describe skills. The *cancel fraction* skill can be represented in clp(FD) as:

```
% Simplify the fraction N/D into its lowest form to give X/Y
% Example:  63/81 gives 7/9
cancel(N,D,X,Y) :-
      domain([N,D,X,Y,F], 1,99),
      F*X #= N,
      F*Y #= D,
      maximize(labeling([], [F,X,Y]), F).
```

Here, variable F is the common factor to be cancelled. This is specified by the two relational constraints. The *maximize* predicate in the final line ensures that the largest value of F will be found.

5 Eliciting the Progression

The knowledge elicitation exercise involved approximately 20 hours of interviews spread over a period of three months. Conventional knowledge elicitation techniques, such as structured interviewing, task analysis and construct theory [24], were used.

Early interviews revealed the significance to the expert of the *skills* that students needed to exercise in order to solve particular problems. The following were identified:

a. Add equivalent fractions
b. Cancel fraction
c. Make proper
d. Find the lowest common multiple
e. Find equivalent fractions

The number of discrete skills required to solve a problem was considered as a measure of the difficulty of the problem; and this measure was used to classify problems, and in so doing reveal a structure of the domain. This coincides with the findings of Beck, Stern & Woolf [19]. However, it is useful to note that this is only one of the many factors in measuring problem difficulty used by Lee [25], who identified, amongst others, the student' s degree of familiarity with a particular type of problem.

In eliciting progression information, it is necessary to avoid the problem of combinatorial explosion. A *head on* approach requires the expert to provide a tree structure of sequences of problems indicating the appropriate *next* problem depending on the outcome of all previously asked problems. Such an approach is unattractive to both expert and knowledge engineer. Instead, an approach adopted was to attempt to uncover the underlying *algorithmic* strategy of the expert.

In general terms, the strategy of the expert is to test the students' abilities to exercise the identified skills at a particular level of difficulty. Failure to return a correct answer causes the questioning process to be resumed at a lower level of difficulty, that is, with problems requiring the demonstration of fewer skills. Whereas successful demonstration of all the identified skills causes the questioning process to be resumed with problems at a greater level of difficulty. The expert started with problems of middling difficulty and adopted a *binary chop* approach to selecting the next level. Within each level of difficulty, the selection of the *next* problem depended on the skills already demonstrated. Each available problem was scored using a set of weights, which favoured previously undemonstrated skills at that level. If the progression problem is viewed as a variant of state-space search, the expert's strategy has more in common with a *constrain-and-generate* paradigm [26], at a given level of difficulty, rather than a naï ve *generate and test* approach. A schematic example of the use of this strategy is given below.

In a Prolog implementation of this strategy, a record of students' skills, demonstrated at each tested level of difficulty is recorded, and used to prepare a revision plan.

6 An Example

The human tutor first prepared the adaptive testing strategy for a domain of five skills described above. This is shown in Figure 1 for a domain of five skills.

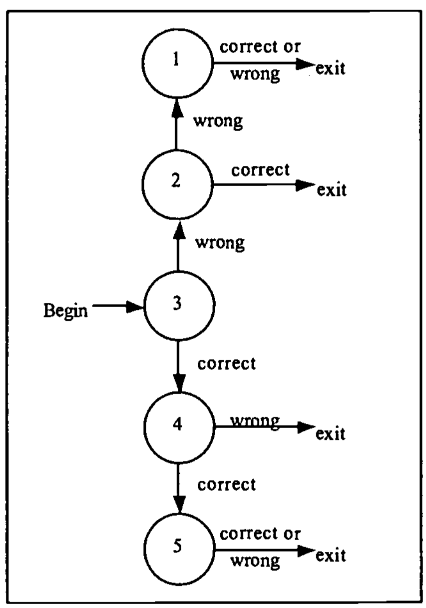


Figure 1: Human tutor' s strategy in adaptive testing for a domain of 5 skills

In Figure 1, the adaptive test begins at node 3 which contains problems each of which can be solved by exactly three skills. If the student gets any problems wrong within that category, he moves onto node 2 which contains problems each of which can be solved by exactly two skills. If he gets all the problems correct within that category, he will exit the adaptive test. The rationalisation for this is described below.

If each of the skills were labelled as a,b,c,d,e, as in Section 5, then at node 3, there are ${}^5C_3$, that is 10 possible combinations of skills. For example, the combination [a,b,c] would involve a set of problems which each require all the skills **a**, **b** and **c** to be used. Skills **a**, **b** and **c** correspond to *add equivalent fractions*, *cancel fraction*, and, *make proper* respectively. However in practice, not all these combinations will be found in a valid problem type.

We introduced weights to each combination to enable the choice of the next best combination. We also imposed the following criteria for calculating the weight of each candidate set:

- If a skill has been not been asked yet, it carries a weight of 2
- If a skill has already been asked once, it carries a weight of 1
- If a skill has been asked more than once, it carries no weight
- Select the first set amongst the candidate set with the highest score

The following process shows how problems, each of which, require a combination of three skills are presented to the student.

a. Select [a,b,c] and scores are assigned to the other combinations, based on the above rules:

	[a,b,c]	[a,b,d]	[a,b,e]	[a,c,d]	[a,c,e]	[a,d,e]	[b,c,d]	[b,c,e]	[b,d,e]	[c,d,e]
1	•	4	4	4	4	5	4	4	5	5

b. Based on these weights, combination [a,d,e] becomes the next best choice and is thus chosen. The scores for the remaining combinations are recalculated.

	[a,b,c]	[a,b,d]	[a,b,e]	[a,c,d]	[a,c,e]	[a,d,e]	[b,c,d]	[b,c,e]	[b,d,e]	[c,d,e]
1	•	4	4	4	4	5	4	4	5	5
2	-	2	2	2	2	•	3	3	3	3

c. Combination [b,c,d] becomes the next best choice and is thus chosen.

	[a,b,c]	[a,b,d]	[a,b,e]	[a,c,d]	[a,c,e]	[a,d,e]	[b,c,d]	[b,c,e]	[b,d,e]	[c,d,e]
1	•	4	4	4	4	5	4	4	5	5
2	-	2	2	2	2	•	3	3	3	3
3	-	0	1	0	1	-	•	1	1	1

d. Combination [a,b,e] becomes the next best choice and is thus chosen.

	[a,b,c]	[a,b,d]	[a,b,e]	[a,c,d]	[a,c,e]	[a,d,e]	[b,c,d]	[b,c,e]	[b,d,e]	[c,d,e]
1	•	4	4	4	4	5	4	4	5	5
2	-	2	2	2	2	•	3	3	3	3
3	-	0	1	0	1	-	•	1	1	1
3	-	0	•	0	0	0	0	0	0	0

e. As there are no more candidate sets, no more problems are presented.

The above example shows that out of the ten combinations, only problems of combinations [a,b,c], [a,d,e], [b,c,d] and [a,b,e] were chosen. As described previously, the human tutor would consider the student's previous performance and if any answers to problems were found to be wrong, he would assign problems at node 2 (see Figure 1). Conversely, if all the answers were found to be correct, he would assign problems at node 4 which require problems to be solved with exactly four skills.

The human tutor took the view that if a student has already tackled problems of three skills, whether he got them right or not, information gathered in packets of three skills need not necessarily apply to problems involving two skills. He considered that students may become anxious about problems which require more skills, and although some of the skills may well have been demonstrated in *easier* problems, the student may find it difficult to apply them in *harder* problems.

7 Conclusion

The paper describes the development of an adaptive test in the domain of elementary arithmetic, which required two styles of knowledge acquisition. The first is concerned with describing problems and skills, and it is computer-assisted; whereas the second is entirely manual and is concerned with the ordering, or progression, of problems to be posed to the subject of a test. However, based on this experience, work is currently underway to develop software to aid with eliciting details of progression. A valuable insight gained is that some degree of formalisation of the problem, as well as being convenient for the knowledge engineer, is also acceptable to the expert who helped with this work.

A possible significant difference between the research reported here and the work reviewed in Section 2 is that the approach to progression is not restricted to a fixed collection of problems. In view of Lee' s findings [25], it would be inappropriate to enforce the equating of difficulty with the number of skills. Evidence encountered during the knowledge acquisition experience suggests that the sheer clerical complexity of mapping out sequences of problems, lead to some draconian simplification on the part of the expert. The task ahead, is to find an appropriate balance between convenience and efficiency.

References

[1] Vispoel, W. P., Rocklin, T. R., & Wang, T., "Individual differences and test administration procedures: a comparison of fixed-item, computerized-adaptive, and self-adapted testing", Applied Measurement in Education, 7, pp. 53-79, (1994).

[2] Collins, J. A., Greer, J. E., & Huang, S. X., "Adaptive assessment using granularity hierarchies and Bayesian nets", Proceedings of the Third International Conference on Intelligent Tutoring Systems (ITS'96), Montréal, Canada, June 12-14, pp. 569-577, (1996).

[3] Dowling, C.E. & Kaluscha, R., "Prerequisite relationships for the adaptive assessment of knowledge", Proceedings of the Seventh World Conference on Artificial Intelligence in Education (AI-ED'95), Washington DC, USA, August 16-19, pp. 43-50, (1995).

[4] Hirashima, T., Niitsu, T., Hirose, K., Kashihara, A., & Toyoda, J., "An indexing framework for adaptive arrangement of mechanics problems for ITS", IEICE Transactions on Information and Systems, E77-D, pp. 19-26, (1994).

[5] Khuwaja, R. & Patel, V., "A model of tutoring: based on the behavior of effective human tutors", Proceedings of the Third International Conference on Intelligent Tutoring Systems (ITS'96), Montréal, Canada, June 12-14, pp. 130-138, (1996).

[6] Wainer, H., "Computerized adaptive testing: a primer", NJ: Lawrence Erlbaum Associates, (1990).

[7] Rios, A., Millan, E., Trella, M., Perez-de-la-Cruz, J.L.,& Conejo, R., "Internet based evaluation system", Proceedings of the Ninth International Conference on Artificial Intelligence in Education (AI-ED'99), Le Mans, France, July 19-23, pp. 387-394, (1995).

[8] Huang, S. X., "A content-balanced adaptive testing algorithm for computer-based training systems", Proceedings of the Third International Conference on Intelligent Tutoring Systems (ITS'96), Montréal, Canada, June 12-14, pp. 306-314, (1996).

[9] Doignon, J. P. & Falmagne, J. C., "Spaces for the assessment of knowledge", International Journal of Man-Machine Studies, 23, pp. 175-196, (1985).

[10] Hockemeyer, C. & Dietrich, A., "The adaptive tutoring RATH: a prototype", Proceedings of the International Workshop on Interactive Computer aided Learning (ICL'99), Villach, Austria, (1999).

[11] ALEKS 2000, http://www.aleks.com (accessed on August 15 2000).

[12] Dowling, C. E., Hockemeyer, C., & Ludwig, A. H., "Adaptive assessment and training using the neighbourhood of knowledge states", Proceedings of the Third International Conference on Intelligent Tutoring Systems (ITS'96), Montréal, Canada, June 12-14, pp. 578-587, (1996).

[13] Villano, M., "Probabilistic student models - Bayesian belief networks and knowledge space theory", Proceedings of the Second International Conference on Intelligent Tutoring Systems (ITS'92), Montréal, Canada, June 10-12, pp. 491-498, (1992).

[14] Kambouri, M., Koppen, M., Villano, M., & Falmagne, J. C., "Knowledge assessment - tapping human expertise by the query routine", International Journal of Human-Computer Studies, 40, pp. 119-151, (1994).

[15] McCalla, G., Greer, J., Barrie, B., & Pospisil, P., "Granularity hierarchies", Computers & Mathematics with Applications, 23, pp. 363-375, (1992).

[16] Hirashima, T., Kashihara, A., & Toyoda, J., "Toward a learning environment allowing learner-directed problem practice - helping problem-solving by using problem simplification", Proceedings of the Third International Conference on Intelligent Tutoring Systems (ITS'96), Montréal, Canada, June 12-14, pp. 466-314, (1996).
[17] Matsui, T., "Optimization method for selecting problems using the learner's model in intelligent adaptive instruction system", IEICE Transactions on Information and Systems, E80D, pp.196-205, (1997).
[18] Carberry, S. & Clarke, J. R., "Generating clinical exercises of varying difficulty", Proceedings of the Sixth International Conference on User Modeling (UM97), Ch ia Laguna, Sardinia, Italy, June 2-5, pp. 273-275, (1997).
[19] Beck, J., Stern, M., & Woolf, B. P., "Using the student model to control problem difficulty", Proceedings of the Sixth International Conference on User Modeling (UM97), Chia Laguna, Sardinia, Italy, June 2-5, pp. 277-289, (1997).
[20] Payne, S. J. & Squibb, H. R., "Algebra mal-rules and cognitive accounts of error", Cognitive Science, 14, pp. 445-481, (1990).
[21] Lightfoot, J. M., "Expert knowledge acquisition and the unwilling expert: a knowledge engineering perspective", Expert Systems, 16, pp. 141-147, (1999).
[22] Clp(FD) 2000, http://www.sics.se/isl/sicstus/docs/latest/html/sicstus.html#CLPFD (accessed on August 15 2000).
[23] Chua Abdullah, S. & Cooley, R. E., "Using constraints to develop and deliver adaptive tests", Proceedings of the Fourth International Conference on Computer Assisted Assessment (CAA), June 21-22, Loughborough University, UK, (2000).
[24] McGraw, K. L. & Harbison-Briggs, K., "Knowledge acquisition, principles and guidelines", NJ: Prentice-Hall, (1989).
[25] Lee, F. L., "Electronic Homework: an intelligent tutoring system in mathematics", PhD Thesis, The Chinese University of Hong Kong, (1996).
[26] Marriott, K. & Stuckey, P., "Programming with constraints: an introduction", Cambridge, MA: MIT Press, (1998).

Development and Evaluation of a Mental Model Forming Support ITS

- the Qualitative Diagnosis Simulator for the SCS Operation Activity-

Toru Miwata, Tatsunori Matsui, Toshio Okamoto and Alexandra Cristea
University of Electro-Communications, Graduate School of Information Systems
Choufu, Choufugaoka 1-5-1, Tokyo 182-8585
Tel: +81-424-43-5621
Fax: +81-424-89-6070
E-mail:matsui-t@ai.is.uec.ac.jp

In this study, we built an educational qualitative diagnosis simulator, which models SCS (Space Collaboration System: system the remote conferences and education via satellite communications) conferences. A student engages in the conference, by operating a control panel and proceeds by making the necessary selections according to the agenda of the virtual conference, and its intention and purpose, which can change at any time. The purpose of this study is supporting the student to form a correct mental model in this environment. Therefore, we incorporate an abstract model of possible computations as a logical circuit attached to the SCS system. Using this model, the system has two functions: to diagnose the student' s conceptual understanding mistakes about the SCS system and to explain to him/her the cause of these mistakes. With these functions, we expect to be able to support the student in forming a correct mental model and in understanding the SCS essentials.

Keywords: **Mental Model, Space Collaboration System, Remote Conference**

1 Introduction

Recently, with the increased awareness of the necessity of individual, subjective learning, a change occurred in the building of computer based educational systems. The existing learning supporting systems are based on automatically generating the learning method, according to the relation between the state defining parameters and the subject' s (learner' s) behavior. However, in recent years, the trend to construct systems, that positively encourage the student to work, and allow him/her to change the current state parameters by him-/herself, offer system behavior simulation, moreover, verification and correction of the student inputs, emerged. In this type of subjective/ individual learning environment, it is necessary to add a causality explanation function of the target environment. This is important due to the fact that, by letting the student/ learner adjust and change the system parameters, and then showing him/her the system behavior simulation, as derived from the current configuration and structure, fundamental system comprehension can be supported and achieved [2..11]. We have, therefore, used the above mentioned specifications and background information, to implement an educational qualitative diagnosis simulator, for supporting fundamental system comprehension and understanding. For this purpose, we have based our mental model design on the object oriented approach. The mental model is a representation of the individual comprehension about the structure and functions of the objects involved in the simulated system model. Moreover, depending on the simulation of the object functions within the learner' s mental model, it becomes possible to predict the problem solving act results. Therefore, important learning can occur and, at the same time, causality explanation within the virtual learning environment can be offered. We based the mental model used in our system on the qualitative modeling. The qualitative model is a fundamental model representation based on the causality relations that generate the target system' s behavior. The causality relations are reflected in the relations between the system' s structure, behavior and functions. Here we consider the following definitions. The **structure** reflects how the elements of the target organization are combined. The **behavior** shows how the system characteristics, expressed by the object structure, change in time. The **function** expresses how the goal, related to the object behavior, is achieved. By modeling the causality relations between the system' s structure, behavior and functions, and designing a qualitative model, the causality relation simulation becomes possible. In our system, we have constructed a qualitative diagnosis simulator for conferences via SCS. SCS, standing for Space Collaboration System, is a remote conferences and distance education system via satellite communications. The learner/ student follows the progress of the conference, by operating a control panel, and making the necessary selections, according to the agenda of the virtual conference, and its intentions and purpose, which can change in time. In this environment, we integrate a computable model abstraction of the remote conference via communication satellites, as a logic circuit. Moreover, based on this abstraction, we add a causality explanation function, and a diagnosis system of the student' s/ learner' s operation mistakes, which generate the appropriate guidance information for the student. In this way, we support the fundamental comprehension of the SCS system.

2 Qualitative reasoning

Qualitative reasoning is one of the most vigorous areas in artificial intelligence. Over the past years, a body of methods have been developed for building and simulating qualitative models of physical systems (bathtubs, tea kettles, automobiles, the physiology of the body, chemical processing plants, control systems, electrical circuits, and the like) where knowledge of that system is incomplete. Qualitative models are more able than traditional models to express states of incomplete knowledge about continuous mechanisms. Qualitative simulation guarantees to find all possible behaviors consistent with the knowledge in the model. This expressive power and coverage are important in problem-solving for diagnosis, design, monitoring, and explanation. Qualititative simulation draws on a wide range of mathematical methods to keep a complete set of predictions tractable, including the use of partial quantitative information. Compositional modeling and component-connection methods for building qualitative models are also discussed in detail [1].

3 SCS

Figure 1 displays the SCS based remote conference concept. SCS was established as a satellite communication network between universities, to enable real-time remote video conferences. Each participant' s station (called VSAT station) is enabled with a satellite communication control panel, an image and sound transceiver control panel, multiple video-cameras, monitors, and so on.

3.1 SCS constrains and limitations

The SCS conference can take place as an inter-station, bi-directional communication between two stations, or as a multiple VSAT stations communication, where only one station has the role of the moderator, and has authority upon transmission control. In the latter case, all the other station, with the exception of the moderator station, are called client stations, and can participate as such in the conference. The moderator station is decided in advance, before the actual conference, by the conference organizer, according to the requested time-schedules and conference contents. The line control is usually under the sole authority of the moderator station. However, a client station can send a request for line usage for transmission to the moderator. This operation is enabled by the proposal request button existent on each VSAT station panel. By pushing this button, a proposal request notification is sent to the control panel on the moderator station. Moreover, during the conference, it is possible for two different stations to send image and sound, namely, the carrier, at the same time, so there can be up to two distinct proposing stations. The respective client stations are depicted in the lower part of figure 1.

The communication satellite has two reception parts, and a converting switch that allows the selection of the received carrier. Depending on the existing constrains and conditions, a decision mechanism is involved, before actually sending the carrier selection from the satellite. After verifying the current constrains and conditions, the carrier is sent from the satellite. This carrier is sent without exception to all client stations. In figure 1, the sending of the carrier to all the client stations is depicted. The station carriers depicted in figure 1 as a black solid arrows show the connection between the individual stations and the transmission part of the satellite. The figure shows also that the satellite receives only two carriers at a time. However, as all stations are connected with the satellite, as depicted by the solid black arrows, all stations are prepared to send a carrier.

The satellite reception part is built of a receptor, and a converting switch. In this way, by means of the restrictions set by the converting switch receptor, the satellite can receive, all in all, only two carriers. Moreover, these have to be from two distinct stations only. Also, in the case of multiple carrier reception, the moderator station operator can decide, according to his/ her free will, to commute to the receiving of one carrier only, disregarding the choices and modes of the client stations. These constrains, limitations and specifications, and the fact that the client stations can all in all send only two carriers, are depicted in the figure as dotted thick arrows. The two carriers that can be sent are named [send 1] and [send 2]. Their contents is re-sent from the satellite. The restriction that the two carriers, [send 1] and [send 2], should not come from the same station is enforced before this re-transmission. Only when al the above restrictions are fulfilled, can the received carriers be broadcasted from the satellite to all stations. At the reception of the broadcast signals, each client station can separate the two carriers, [send 1] and [send 2]. The station sending the carrier is also receiving the broadcast, without exception. Therefore, the sound and image received by the transmitting stations are:

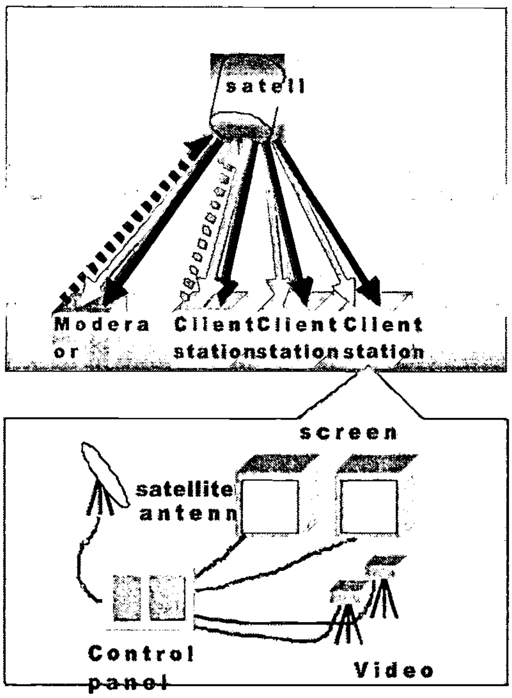


Fig. 1 Conference with SCS

(1) image+sound from the other transmitting station (if existent);
(2)the image and sound sent to the satellite by the station itself.

Moreover, as it is impossible to send the image and sound carrier to a specific station directly, by sending them to the satellite, they are broadcasted automatically to all stations. Bi-directional communication is also possible, but is actually a quasi-bi-directional communication, as the broadcast carrier of the two communicating stations is sent, at the same time, as a broadcast signal to all client stations.

3.2 SCS system frequent user errors

In table 1, the error types for different user skill levels of SCS conference practice, as gathered by surveying 4 domain specialists with over 2 years of SCS system operation experience, is shown. They were asked to give us first a list of frequently appearing user errors during the SCS usage and managing. This list is displayed in table 1 in the column headed by the label "Error/ misconception". Next, they were asked to evaluate the frequency of apparition of these errors for beginner, medium and advanced user. In table 1 their replies were represented as follows: [•] means high, [• •] means medium, and [• •] means low frequency of errors. The table presents therefore the specialists' primary classification of errors according to the operation skills. To this classification, we have added a new error classification, based on the previously explained SCS system constrains and limitations. We have managed to group all errors enumerated by the specialists into four big classes of errors and misconceptions: A, B, C and D. The definitions of these classes are given below.

Table 1 Error types

Error/ misconception	beginner	medium	advanced	Error classification
Disregarding the function of the satellite• • believing direct/ dedicated transfer between fellow stations is possible.	• •	• •	• •	A,B,C, D
Believing that the sending of two carriers from the same station is possible.	• •	• •	• •	A
Believing that receiving two carriers from the same station is possible.	• •	• •	• •	A
Not understanding that, by switching the carrier to a different station, the current proposing	• •	• •	• •	B

station carrier will disappear.				
Believing that all stations can send a carrier at the same time.	• •	• •	• •	C
Not understanding the concept and necessity of the carrier request proposal.	• •	• •	• •	C
Assigning carriers to three or more stations.	• •	• •	• •	C
Not understanding why the image and sound signal sent by ones own station is received again.	• •	• •	• •	D
Believing that bi-directional communication is possible only with a specific station.	• •	• •	• •	D
Assigning the carrier to each station consecutively.	• •	• •	• •	D
Believing that only one broadcast is possible.	• •	• •	• •	D
Not making the distinction between the moderator station and the other client stations.	• •	• •	• •	static state
Believing that the client station is in charge of the transmission control.	• •	• •	• •	static state
Not making the distinction between the two wave forms (signals), [send 1] and [send2].	• •	• •	• •	static state
Believing that the [send 1] wave form is the signal coming from the moderator station.	• •	• •	• •	static state

Class A: Misconception/ incomplete information about the sending of two different waves/ signals with the help of the judgement/ decision mechanism.
Class B: Misconception about the sending of one carrier to one station with the help of the converting switch.
Class C: Misconception/ incomplete information about the receiving of two carriers.
Class D: Misconception/ incomplete information about broadcasting to all stations.

4 The SCS qualitative model

Figure 2 shows the qualitative model of the SCS conference abstraction, in the form of a logic circuit. This qualitative model can express the structure, behavior and functions of the SCS system. In this figure, we displayed four client stations and one communication satellite. As can be seen, the satellite has two receptors, and one judgment/ decision mechanism, as a converting XOR switch between the two receptors. The two client stations sending carriers at one time can therefore have a pseudo- bi-directional communication. The structure, behavior and functions, so, the objects of the original SCS system are expressed, in this way, as a qualitative model.

The characteristics of this model make it possible to simulate the dynamic changes occurring during a distance conference, allowing to decide and evaluate the proper parameter settings for each station, moreover, to simulate the system behavior in the case of mistaken parameter settings. By using the XOR function, it is ensured that each reception part of the communication satellite can receive only one carrier from only one station. This station has sent a prior transmission proposal to the moderator station, which was accepted.

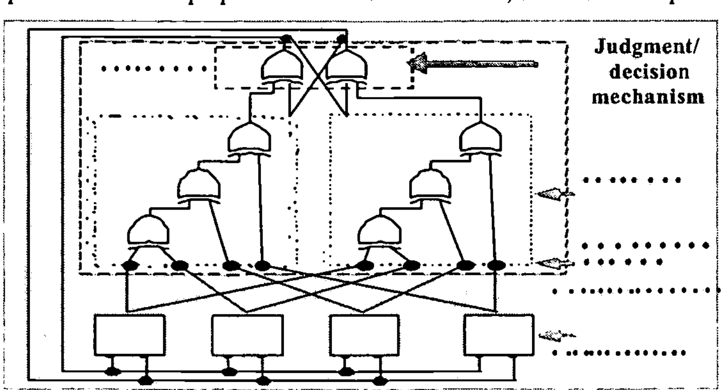
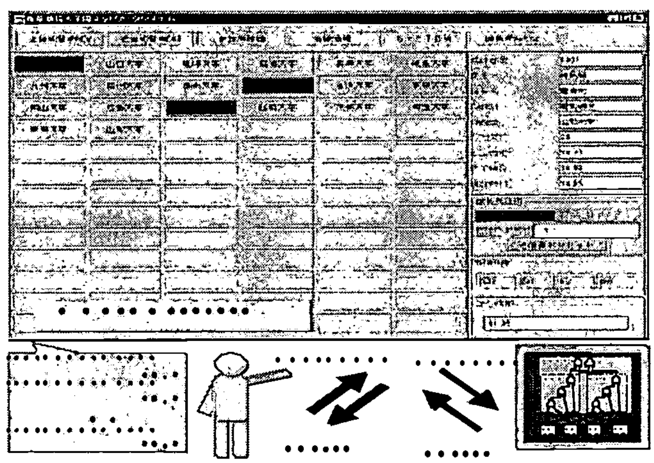


Fig 2 The qualitative model of the • • • •system

Next, it is necessary to make sure that the two accepted carriers come from two distinct stations. This restriction is enforced by the judgment/ decision mechanism. The judgment/ decision mechanism eliminates via an extra XOR function the possibility that the two carriers were sent by the same station. If the two carriers, 1 and 2, are validated by the judgment/ decision mechanism, the communication satellite broadcasts one or both to all VSAT stations. Therefore, all VSAT stations will receive the two carriers 1 and 2 and will not be able to receive any other carriers from other stations, or any wrong transmissions. Moreover, by using this model it is possible to infer the error source, as shown previously, based on the SCS system structure. The previous A, B, C, D classification can be thought of as: (A) sending of two distinct waves by using the judgment/ decision mechanism, (B) sending of maximum one carrier per station by means of the converting switch, (C) using of two carriers by means of the satellite reception mechanism, (D) existence of broadcast type of transmission only. In this way, the virtual model enables the learner to derive the cause and source of the operation error, as related to the SCS system structure. Furthermore, we have presented here a model based on only 4 client stations, that is implemented via the XOR module, but as in the case of more than 4 client stations, we can increase the number of the reception part XOR modules, adapting them to the number of stations, we can express, cope with and model therefore the converting switch for any arbitrary, greater than 2 number of client stations.

5 Learning Environment

5.1 System outline and overview



Dynamic

Figure• •Learning environment

Figure 3 shows the overview of the system. The learner/ student is performing the conference steps by taking over the role of the moderator station operator. The goal is to cope with the dynamically changing agenda of the conference, proposed by the system. The agenda presents a description of a dynamic conference state, where bi-directional communication is required. The student can take decisions about the SCS system state and change parameter by operating the control panel. The previously described qualitative model evaluates these settings and parameters.

Next, disregarding if the parameter setup and assignment is appropriate or not, the result of the new user choices is reflected on the control panel of the interface, changing the current representation. The control panel displays also the transmission requests coming from other stations. The student has to choose the appropriate response to these requests. The student has to be able to judge the appropriateness of his/her own operations and actions, by interpreting the information presented on the control panel. By repeating the above steps, the student can learn the constrains and usage of the SCS system. Moreover, to prevent deadlock situations, where the student is unable to judge his/her own errors, due to misunderstandings regarding the SCS system constrains, an explanatory function was added. This is implemented via an explanation button, which can be pressed by the student in need. The student guidance follows as has been previously shown, conform with the SCS qualitative model. In this way, the student can achieve not just a quick, superficial understanding, but also a deep, structure related knowledge about the SCS system. For example, explanation are given such as: "There are only two satellite receptors.", "There is an exclusive OR switch on each receptor, so each receptor can receive from one only station at a time.", "The judgment/ decision mechanism does not allow 2 carriers from the same station.", and so on. By leading the student to understand the connection between the parameter setup and the way the SCS system is actually built, as well as the real system components and the relations between them, via messages and state representations on the control panel, the student can be expected to perform the parameter setting by him/herself successfully in the future.

5.2 System flow

Figure 4 shows the system flow. The rapidly changing conference goal and intention of the agenda is described in chronological order. The contents of this description are on one hand, the conference state change requirements that have to be performed by the student, put into words that can be easily understood by him/her, and on the other hand, the description of the current SCS system state. In figure 4, this is expressed as [word] utterances, at the different moments in time (t0, . ., tn):

word : state(t0) ~ word : state(tn)

For example, [word] can be a prompting message about the conference state change, with the value of "Please reply to the question from university A!", and so on. As shown in figure 4, the operation panel managing module receives from the agenda, or from the other client stations the current parameter for each given conference state, and then reflects the resulting state on the panel. For example, the button of the station, which is currently in charge of a carrier, turns red. Also, in the case of requests from other stations, the button of the station sending the carrier request signal turns also red.

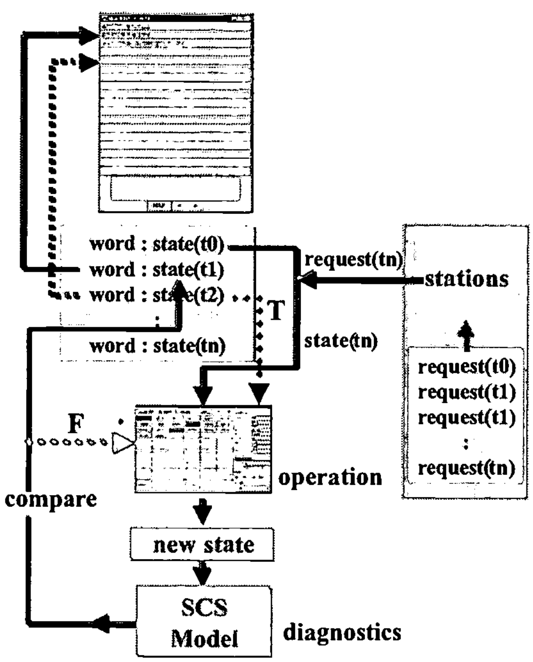


Figure 4 System flow

The student infers the present conference state from the state of the panel. Moreover, from here the student can notice if it is necessary to change the state of the conference, according to the agenda requirements. Next, to change the conference state, the student has to operate the control panel. By doing this, the parameters determining the conference are changed, and a new conference state emerges. This new state is evaluated with the SCS qualitative model. When evaluating with the SCS model, the result is compared with the next agenda. It is, in principle, possible to perform such comparisons on the SCS system without the computable module, and to judge if the operation is appropriate or not, but, in that case, the student cannot achieve a deep understanding of the SCS conference, that is, s/he cannot identify the SCS behavior as derived from structural constrains. In order for the learner to achieve a deep understanding, it is necessary to perform the parameter evaluation with the help of the SCS computable model. After the parameter evaluation, if the settings are judged as appropriate, the system moves to the next agenda. In figure 4, this is the case of "T" (True). In this case, the setup parameters decided by the student are handed over to the administrating module, which, in turn, reflects these changes on the operation panel. On the other hand, if, after the parameter evaluation, the settings are judged as not being appropriate, the system does not move to the next agenda. This case is shown in figure 4 as the "F" (False) case. In such a case, the wrongly set parameters are displayed on the operation panel. In this way, the deficient, real SCS state can be represented.

For example, in the case when three or more stations ask for the carrier at the same time, and the carrier is passed over to them, the moderator station's carrier disappears. The student notices that the respective state is not appropriate, and corrects the setup parameters. Moreover, in the case that s/he doesn't notice the errors, s/he cannot continue with the next agenda. When entering a deadlock situation, the SCS qualitative model can, at the student's request, explain to the student what kind of error s/he has done. In this way, by explaining not the protocol and process steps, but the SCS system behavior, as a result of the structural constrains, our system supports the formation of the SCS learner mental model. For instance, let us consider a case where the present transmission rights belong to universities B and C, and a proposal request is received from university A. This request is represented on the panel by the button representing university A turning red, together with a simultaneous indication message appearing in the agenda window, stating "Please answer the question from university A". If the student decides to assign a carrier to university A, without previously modifying the state of one or both stations B and C, which have the current transmission rights, the result is that the system will have 3 or more simultaneous carriers at the same time. In this case, the system represents the buttons of universities A, B, C on the panel with red color, and lets the student therefore know that the parameter setup is not appropriate.

At the same time, the agenda window will also display a message for the student. The content of this message is something like: "There are only two receptors on the satellite.", so is an explanation of the behavior, as resulting from the structural constrains.

6 Agenda

Table 2 Agenda example

agenda(t0)	The conference starts.
agenda(t1)	The moderator station is the University of Electro-Communications.
agenda(t2)	Please allocate carrier • •to Yamagata University• •
agenda(t3)	Please start sending from the lecturer camera• •
request(t4)	Carrier request • Tsukuba University• •
agenda(t4)	Please reply to the question from Tsukuba University• •
agenda(t5)	The conference has ended• •

The SCS conference is based on a general agenda. Our system offers SCS based remote conference simulation environment and, moreover, stores typical SCS agenda models, in order to dynamically produce conferences that require conference state changes.

In this way, the student becomes the operator of the moderator station, and has to take decisions compatible to the agenda, engaging therefore in the simulated steps of the SCS conference. In table 2 we show an example of a model agenda for our system. In this table, agenda(tn) represents the agenda at moment (tn) in time, and request(tn) represents the carrier request at moment (tn) in time. In the real SCS conference, the time moment concept exists, but, in our system, we have the supplementary restriction that, only after accomplishing the current agenda, it is possible to go on with the new one. As shown above, the agenda is organized as a time series, and the student receives indications and instructions from the agenda window. The changes occurring in the conference state in the respective agenda example above correspond to a respective intention and goal. Disregarding if these intentions and goals come from the original operator's decisions, or if they were prepared by the system from the beginning, the beginner student doesn't have to loose his/her way during the SCS conference proceedings, and can give the panel operation his/her undivided attention. In other words, the indications and instructions coming from the agenda window can be thought of as an experienced operator teaching the beginner student during the SCS conference proceedings. After receiving the indications and instructions from the agenda window, the student can decide on the next conference state that seems appropriate, given the present conference state and the indications received, and operates the control panel to perform the respective change. The new state that results as a consequence of the student's operations is checked by the system, to decode if it is appropriate or not, conform with the indications and instructions of the agenda. One agenda is recorded in the system as one word and 6 state descriptors. The words are the ones that appear in the agenda window. The six possible state descriptors are shown below.

- *station name (list of all client stations)*
- *carrier request (list of all client stations)*
- *carrier 1 (list of all client stations)*
- *carrier 2 (list of all client stations)*
- *reception 1 (list of all client stations)*
- *reception 2 (list of all client stations)*

The state descriptor called "station name" contains a list of all client station names. Next, the carrier request, carrier 1, carrier 2, reception 1 and reception 2 state descriptors contain respective lists of [on] and [off] states corresponding to each station. In figure 3, we show the correspondence between [1] and [0] and [on] and [off]. The reason of describing all client stations carrier and reception states with [off/on] descriptors is to be able to represent also the incomplete understanding of the learner/ student, as well as his/her mistaken parameter setups and assignments.

7 Testing, experiments and evaluation

Table 3 comparison of situation before and after learning takes

Error	Pre Test	Post Test	Error classification
Disregarding the function of the satellite and believing direct/ dedicated transfer between fellow stations is possible.	N/A	N/A	A,B,C,D
Believing that sending of 2 carriers from one station is possible.	3 persons	1 person	A
Believing that receiving two carriers from the same station is possible.	3 persons	1 person	A
Not understanding that, by switching the carrier to a different station, the current proposing station carrier will disappear.	5 persons	1 person	B
Believing that all stations can send a carrier at the same time.	3 persons	3 persons	C
Not understanding the concept and necessity of the carrier request proposal.	5 persons	1 person	C
Assigning carriers to more than three stations.	5 persons	1 person	C
Not understanding why the image and sound signal sent by ones own station is received again.	5 persons	3 persons	D
Believing that bi-directional communication is possible only with a specific station.	5 persons	1 person	D
Assigning the carrier to each station consecutively.	5 persons	1 person	D
Believing that only one broadcast is possible.	5 persons	5 persons	D

We have performed an evaluation experiment of our system over a small sample. 5 beginner students with no SCS system experience were selected as the object of our SCS conference experiment. We have first explained them the control panel representations, meanings and operation mode, as well as the agenda window functionality, and the SCS system setup as a bi-directional communication system. They were able to consult the SCS user manual. Next, we have done a pre-test with the system without the diagnosis mechanism, and followed and checked the operations and mistakes of the beginner operator. Then, we have performed the same experiment, this time, with the help of the diagnosis mechanism. In the last step, we have compared the understanding level before and after learning. The result is displayed in table 3. A system screen display during the experiment is shown in figure 3. This figure displays a student deadlock situation, where the student has asked for an explanation about the deadlock, and the system has next checked the SCS system structure related error cause, and finally displayed it on the screen for the student to see. In the case presented in figure 3, the student hasn't realized the fact that there are only two receptors on the satellite, and has mistakenly allocated carriers to 3 stations. The explanation of his/her error is displayed on the control panel. The state of 3 stations having the carrier is represented on the panel as the respective stations' buttons turning all red (left corner of fig. 3, darkened buttons). However, if the student doesn't grasp the meaning of the representation and the cause and source of his/her errors, and asks therefore the system for help, the system will display the following message: "There are only two receptors on the satellite". With this explanation, the student understands that, as there are only 2 receptors on the satellite, s/he cannot allocate carriers to 3 stations, and will operate the panel correctly in his/her next steps.

According to our system's result shown in table 3, the students can understand the SCS system constrains and limitations, the fact that the signal has to be sent from different stations, the fact that there are only two carriers, and the concept of the XOR receptors of the satellite. However, the broadcasting mechanism was not completely understood. This is probably due to the fact that, in the current simulation system, there is no visual display of the broadcasting mechanism, of the time and direction of the transmission.

7 Conclusion

In this paper, we proposed an educational qualitative diagnosis simulator based on an object-oriented approach to mental model formation. In our model, the structure, behavior and functions of the SCS system are the objects, and from the description of the causality relations between these objects, the student can determine the cause of his/her error, based on system structure judgment.

From educational strategy point of view, QUAD implements and supports a combination of learning methods, like "Reinforcement learning", "Learning by exploring", "Learning by asking", "Learning by applying", "Self-monitoring", and so on. From educational depth point of view, the QUAD system doesn' t stop at the procedural surface level, but traces the structural implications, to gain a deep knowledge level.

For further research, we believe that, by expanding the current system, and identifying more precisely the mental model of the student, a more appropriate guidance system can be developed.

References

[1] B. Kuipers; Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge, Cambridge, MA: MIT Press (1994).

[2] T. Okamoto• O. Matsuda; About the mental model of the ITS based geometrical/ topological proof process, Japanese Journal of Education and Technology,Vol.4, No.15, pp.167-182 (1992)

[3] O. Matsuda, T. Okamoto; based on hypothesis inference/ deduction : about the ITS - student model and its recognition method, IEICE conference, A, Vol. J75-A, No.2, pp.266-274 (1992)

[4] T. Okamoto; Knowledge Processing Technology and Educational Engineering, Japanese Journal of Educational Technology 20(4), pp. 207-226, (1997)

[5] T. Nishida; The basis of qualitative inference/ deduction, Journal of the AI society, Vol.4,No.5,pp.522-527,(1989)

[6] T. Okamoto; Education, Media and Artificial Intelligence, Journal of the AI society, Vol.10,No.3,pp.361 – 367 (1995)

[7] T. Okamoto; Knowledge Processing Technology and Educational Engineering, Journal of the AI society, Vol.20, No.4, pp.207 – 226 (1997)

[8] T. Nishida; The common sense thinking way model of qualitative inference/ deduction• •Journal of the AI society, Vol.2, No.1, pp.30 - 43,(1989)

[9] T. Nishida; Towards applications of the thinking ways and intelligent problem solving techniques of qualitative inference/ deduction• •Journal of the AI society, Vol.32, No.2, pp.105 • •117 (1991)

[10] T. Nishida; Aspects of qualitative inference/ deduction, Asakura publishers (1993)

[11] K. Asai, H. Ohnishi, K. Kondo, K. Tanaka, O. Matsuda; Case base and Collaboration Application based on appropriate• • •(control) and the Internet system, technical Report of IEICE, ET97-24 (1997-05)

Intelligent Interactive Learning Environment: Design Issues

Siu-Cheung KONG* and Lam-For KWOK**
**Department of Information & Applied Technology, Hong Kong Institute of Education, 10, Lo Ping Road, Tai Po, N.T.,Hong Kong. Email: sckong@ied.edu.hk*
***Department of Computer Science,City University of Hong Kong,Tat Chee Avenue, Kowloon, Hong Kong. Email: cslfkwok@cityu.edu.hk*

Interactive Learning Environment (ILE) provides interaction opportunities between learners and the virtual devices for productive learning. Intelligent ILE (IILE) provides quality feedback or authentic guidance to learners who need help in the ILE. This research aims to explore design implications of IILE by studying model of learner in the mathematics fraction domain. 169 primary four learners were invited to answer 10 open-ended questions on fraction addition and subtraction. A learner model on category of error and error pattern was formulated from the 423 erroneous responses. Results of the study indicated that researchers should study error patterns by understanding work of learners, distinguish careless mistakes from error patterns, and consider scaffolding support.

Keywords: **Intelligent Interactive Learning Environment, Learner Model**

1 INTRODUCTION

There are two categories of Learning Environment (LE): content-free and subject-specific [1]. A content-free LE allows participants and facilitators to formulate their own topics for discussion. Knowledge formulated from such interactions belongs to the learning community [2]. A subject-specific LE involves subject knowledge. Some subject-specific environments stress knowledge transfer like Intelligent Tutoring System (ITS) [3]. Other subject-specific environments such as Interactive Learning Environments (ILE), assisting learners to learn through exploration, put efforts on designing manipulative virtual learning devices [4]. No matter an LE is designed for knowledge transfer or knowledge formulation, subject matter of the learning domain should be carefully studied and incorporated in it [5].

1.1 Design Considerations of an ILE

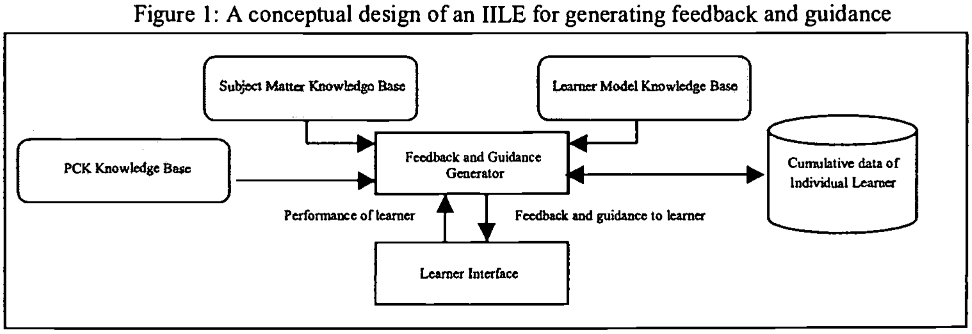
The study of subject matters plays a crucial role in designing ILE involving knowledge exploration because learners are not obtaining knowledge directly from the ILE. Learners have to learn by analogy, that is, learners have to transfer knowledge from manipulating the manipulative virtual devices of the ILE to grasp the abstract concepts of the subject domain [4]. Expert teachers are skilful in predicting how learners will think and err [6]. This diagnostic ability is tied to an expert's special understanding of the subject and is undoubtedly derived from multiple opportunities to teach the same content [7]. This knowledge includes knowing which aspects of a topic are particularly difficult, what the common misconceptions are, and what representations are important for authentic learning. Shulman [8] termed this kind of knowledge as Pedagogical Content Knowledge (PCK). It is crucial to utilize teachers' expert knowledge, especially knowledge on representation for authentic learning, to design manipulative virtual devices of an ILE.

1.2 Design Considerations of an Intelligent ILE

An ILE may provide interaction opportunities between learners and the virtual devices for productive learning. Some learners may learn the subject matter well without the assistance of the virtual learning devices. Some learners may learn well with chances to interact with the interactive learning devices of the environment. However, some learners may need guidance to learn well in the ILE [9]. An Intelligent ILE (IILE) is an ILE that provide feedback or guidance to those learners who need such help in learning the subject domain. Those learners who do not need help will not notice the existence of the auxiliary service. Learner model of learning in a subject domain may provide information about the behaviour of learners in learning the domain. Studying the learning model of learners may assist IILE designers to formulate design principles and obtain technical details such as formulating mal rules for understanding learning states of learners. A learner model thus may help to tailor-make an IILE for assisting various types of learners in learning the discipline. It is therefore important to study the learning model of learners in a specific subject domain for designing a useful and practical IILE to assist learners of various kinds in the learning process.

Three knowledge bases are therefore important for designing an IILE for learning subject-specific knowledge. They are the subject matter, the learner model of learning in the domain and the PCK of teachers in teaching the discipline. Subject matter knowledge base contains subject matter knowledge. It can provide subject matter advice and knowledge state of learners in the learning process. Learner model contains behaviour representations of learners. Learner model knowledge base may provide information about the learning state of learner. PCK knowledge base contains diverse guidance knowledge for different learning states of learners. It may provide learning advises based on PCK of experienced teachers of the subject domain who know how learners think and err in the discipline. Software agents will monitor the performance of learner in the learner interface. Software agents will determine proactive or reactive responses after a negotiation and communication process in the feedback and guidance generator. The negotiation will be a judgement of the knowledge state of the learner in the domain using both the learner model knowledge base and subject matter knowledge base of the IILE. Final decision will be an outcome after a consultation with the PCK knowledge base of the IILE and the cumulative data of an individual learner. The cumulative data records the historical learning states of each individual learner captured by the IILE. Figure 1 shows a conceptual design of an IILE for generating feedback and guidance.

Figure 1: A conceptual design of an IILE for generating feedback and guidance



1.3 Chosen Subject Domain

A review of literatures indicated that many learners have great difficulties in learning the concepts and procedural knowledge of mathematics fraction [10, 11, 12]. Streefland [11] further pointed out that the main cause of such difficulties is the inadequate and inappropriate teaching in the traditional approaches. As the teaching and learning of mathematics fraction is an internationally renowned difficult topic, it is considered as an appropriate exemplar to be investigated for automation.

2 AIM AND OBJECTIVES

The aim of this research is to study the knowledge of learners in a subject-specific domain and to investigate its implication for designing a subject-specific IILE. There are two specific objectives: (1) to understand the problems of learners in learning the topic; (2) to discuss design issues of an IILE. Such findings may inform the development of IILE for providing quality feedback and guidance to learners.

3 RESEARCH METHODOLOGY

A questionnaire for studying model of primary learners on learning fraction addition and subtraction was designed.

169 primary four learners from four different schools were invited to complete the questionnaire through their mathematics teachers. All learners had completed their learning of fraction addition and subtraction before the test. Learners were requested to do the questionnaire on individual basis in a mathematics lesson for about 35 minutes. No discussions were allowed. The answer sheets were not used for any form of assessment but returned to the researcher after the administration. All 169 answer sheets returned were used for data analysis.

4 RESULTS AND DISSCSSIONS

This section will report on the quantitative and qualitative analysis results of all errors responded by participants of the survey and will discuss their implications on designing an IILE. The learner model formulated contains two areas: (1) knowledge of learners on category of error; and (2) knowledge of learners on error patterns of the domain.

4.1 Knowledge of Learner on Category of Error

Nine categories of error were identified and summarized from the 423 incorrect responses. Though incorrect response of each question may contain more than one error, this study selected the primary source of error for classification. Results were summarized in table 1. Categories were organized in descending order of percentage that account for the errors. The summarized result may serve as an important reference in designing a learner model of LE for fraction learning. Among the nine categories, categories 1, 2 and 9 directly related to the subject matter and accounted for nearly forty percent of the erroneous work. Categories 3 and 8 were common types of error in any mathematics exercise. It is interesting to investigate whether learners in this age group would commit these types of error like doing subtraction for addition at a certain level of unconsciousness. The study reflected that these factors might account for another twenty percents of errors.

Table 1: Category of error summarized from the learner model of the study

	Category of Error	Percentage Accounted
1	Improper handling of mixed number in fraction operation	20.4%
2.	Insufficient procedural knowledge for evaluating fraction	14.7%
3.	Calculation or careless mistake	13.5%
4	Unable to set up correct expression for solving word problem	11.6%
5.	Incorrect strategy for evaluating expression	11.4%
6.	Unable to identify error pattern for erroneous work	10.9%
7.	Not responding to question or the piece of work unfinished	8.5%
8.	Conducting subtraction for addition and similarly addition for subtraction	5.5%
9.	Incorrect simplification of answer to the simplest fraction form	3.6%

Though categories 4 and 5 can be purposely avoided, they do play a role in mathematics learning. Setting up expression for solving problems in a scenario may help to test whether a learner has grasped the taught concept. Strategies of evaluating numerical expressions may help to detect whether a learner has knowledge on magnitude of operands and order of evaluation on operators in an expression. The deficiency of this knowledge accounted for twenty percents of errors detected in this study. Categories 6 and 7 accounted for the last twenty percent of learners' work that might not be understandable or remain unfinished. Those 10 percent of learners' work could not be identified for any error pattern reflected that even human teachers might be unable to understand open-ended pieces of work like evaluating mathematics expressions.

4.2 Knowledge of Learner on Error Patterns

This section will report on knowledge of learners with problems in working with fractions on addition and subtraction. After careful analysing error patterns of learners in evaluating and solving simple fraction addition and subtraction problems, two categories were summarized: (1) concrete error pattern; and (2) vague idea on working with fractions. The first category includes some concrete error patterns that can be abstracted into mal rules. The second category contains error patterns that cannot be easily summarized into mal rules but reflect vague ideas and incomplete working procedures of learners. One of the most famous mal rules on fraction addition can be named as "Add numerators and add denominators". Learner with poor knowledge on fraction addition will adopt knowledge of arithmetic addition by adding the numerators of fractions in the fraction expression to give the numerator of the resultant fraction and similarly adding the denominators of fractions to give the denominator of the resultant fraction. There were four learners committing this type of error in this study. This rule might explain 3% of the errors. The second category of error pattern to be analysed involves high-level abstraction. The group of learners in this category showed no concrete error patterns. However, the pattern illustrated that these learners have some vague ideas of doing fraction addition and subtraction. Examples were illustrated in table 2.

Table 2 Vague ideas for evaluating fraction addition and subtraction expressions

	Error 1	Error 2
Learner 1 (3 score)	$\frac{3}{8}+\frac{1}{6}=\frac{9}{18}+\frac{9}{18}=\frac{18}{18}=1$	
Learner 2 (6 score)	$\frac{1}{2}+\frac{1}{3}=\frac{3}{10}-\frac{2}{10}=\frac{5}{10}=\frac{1}{2}$	
Learner 3 (0 score)	$\frac{1}{2}+\frac{1}{3}=\frac{1\times3}{2\times3}+\frac{1\times3}{3\times1}=\frac{3}{6}+\frac{2}{6}=\frac{6}{5}$	$\frac{3}{8}+\frac{1}{6}=\frac{3\times6}{8\times6}+\frac{1\times6}{3\times6}=\frac{54}{18}+\frac{3}{18}=\frac{12}{18}=\frac{6}{5}$

These erroneous presentations reflected that learners did have vague ideas about the working procedures on fraction addition. They need assistance to organize the disconnected nodes into a semantic net. Result of the studies indicated that some error patterns could be represented by mal rules. However, there were even more that cannot. An alternate method of studying error patterns of learners is to understand their work.

Identify Careless Mistake

The learner model of this study reflected that twenty percent of errors were derived from calculation or careless mistakes. Careless mistakes in this study mean transcription errors or simple computational mistakes form one step to another. The feedback and guidance will be different if an error is identified as a careless one. An IILE should handle not only problems generated from subject matters but also general problems of learner like careless mistake. An authentic guidance should provide not only advices or actions that can assist learners to formulate conceptual understanding of the subject domain but also offer help to learners derived from general problems such as careless mistakes. An IILE should attempt to distinguish careless mistake from other error patterns like human teachers.

Scaffolding Support

The forty percent of errors derived from inadequate knowledge of learners reflected that only immediate feedback may not help learner much and thus authentic guidance should be considered for facilitating conceptual understanding. A productive learning support should be an arrangement of a sequence of situations for facilitating knowledge construction [12]. The role of a mathematics-learning environment will be to help learners to learn, especially those fundamental concepts in mathematics, but not to replace mathematics learning in the conventional manner. Therefore it is fundamental for such kind of learning environment to provide scaffolding support to learner when assistance is needed. Support should gradually withdraw so that learner can stand on its own after leaving the system. Therefore a fraction IILE should be designed like a blank sheet for learner to work with fraction. Feedback and guidance are only provided when it is needed. On the other hand, learner working in the IILE who does not need support will not notice the IILE in behind.

5 CONCLUSION

Studying the learning model of learners may assist IILE designers to formulate design principles and obtain details for understanding learning states of learners. The learner model of this study modelled behaviour of learners in two aspects: error category and error patterns. Nine categories of error were identified. Forty percent of errors were derived from inadequate knowledge of learners on subject matters. Twenty percent could be explained by careless mistakes. Twenty percent involved general mathematics knowledge. The final twenty percent of erroneous work were difficult to be classified or work was not completed. Learner model of the study reflected that some error patterns could be represented by mal rules. However, there were even more that cannot. An alternate method of studying error patterns of learners is to understand their work. Result of the study indicated that IILE needed to apply a strategy to identify careless mistake so that appropriate guidance to learners can be provided. The forty percent of errors derived from inadequate knowledge of learners reflected that only immediate feedback may not help much and thus authentic guidance should be considered for facilitating conceptual understanding. A productive scaffolding support should be an arrangement of situations for facilitating knowledge construction. The future work of the study is to design ways and means to understand work of students, to devise strategy to distinguish careless mistake from other error patters, and to plan scenarios for assisting learners to learn by exploration in an IILE.

References

[1] Vosniadou, S., Corte, E.D., Glaser, R., & Mandl, H. (Eds.), "International perspectives on the design of technology-supported learning environments", Hillsdale, NJ: Lawrence Erlbaum, (1996).

[2] Duffy, T., & Jonassen, D. (Eds.), "Constructivism and the technology of instruction: A conversation", Hillsdale, NJ: Lawrence Erlbaum, (1992).

[3] Chan, T.W., "An introduction to intelligent computer assisted learning systems research and development", National Science Council Monthly, Vol. 23, No. 5, pp. 456-468, (1995).

[4] Akpinar, A., & Hartley, J.R., "Designing interactive learning environments", Journal of Computer Assisted Learning, Vol. 12, pp. 33-46, (1996).

[5] Wittmann, E.Ch., "Mathematics education as a 'design science' ", Mathematics education as a research domain: A search for identity, Dordrecht: Kluwer Academic Publishers, pp. 87-103, (1998).

[6] Berliner, D.C., "In pursuit of the expert pedagogue. Educational Researcher", Vol. 15, No. 7, pp. 5-13, (1986).

[7] Livingston, C., & Borko, H., "Expert-novice differences in teaching: A cognitive analysis and implications for teacher education", Journal of Teacher Education, Vol. 40, No. 4, pp. 36-42, (1989).

[8] Shulman, L.S., "Those who understand: Knowledge growth in teaching", Educational Researcher, Vol. 15, No. 2, pp. 4-14, (1986).

[9] Cheng, C.C., "Proactive guidance in computer-assisted language learning", Technical Report, LLL-T-24-95. Language Learning Laboratory, College of Liberal Arts and Sciences, University of Illinois, (1995).

[10] Behr, M.J., Harel, G., Post, T. & Lesh, R., "Rational number, ratio and proportion", Handbook of research on mathematics teaching and learning, New York: Macmillan, pp. 296-333, (1993).

[11] Streefland L., "Fractions in realistic mathematics education: A paradigm of developmental research", Dordrecht: Kluwer Academic Publishers, (1991).

[12] Ohlsson, S., "Knowledge requirements for teaching: The case of fractions", Teaching knowledge and intelligent tutoring, Norwood, NJ: Ablex Publishing Corporation, pp. 25-59, (1991).

Microgenetic Analysis of Conceptual Change in Learning Basic Mechanics

Gary Chon-wen Shyi and Shih-tseng Tina Huang
Department of Psychology and Center for Research in Cognitive Science
National Chung-Cheng University
160 San-Hsing, Ming-Hsiung, Taiwan 621, R. O. C.
voice-mail: +886 5 2720411 ext. 6102
e-mail: psycws@ccunix.ccu.edu.tw

Microgenetic approach to understanding the process of cognitive development entails repeatedly assessing participants' performance on a conceptual domain undertaking rapid change. In the present study, we adopted the microgenetic method to examine the conceptual change process in learning elementary Newtonian mechanics. Twelve junior-high school students with comparable competency in mechanics were assigned to two groups, and their understanding of concepts in elementary mechanics were assessed in four occasions by interacting with a computerized test-bank software. Participants in the group-test condition were assessed in a group setting. Participants in the individual-test condition were additionally asked to provide explanations for their answers to each test item. The results showed that while participants benefited from being repeatedly tested and showing a increasingly higher level of sophistication in their understanding for most of the conceptual domains tested, it doesn't matter whether or not they also offered explanations to their own answers. More importantly, the differences in developmental course across conceptual domains and the variability of developmental course within conceptual domains together lend support to the theoretical assumptions of the microgenetic approach.

Keywords: **conceptual change, microgenetic analysis, computer-assisted testing, basic mechanics**

1 Introduction

There is little doubt that one of the major challenges in understanding cognitive development is to have an adequate account for the process of conceptual change. Over the last few decades, research in cognitive development has produced a number of distinctive approaches to understanding the process of conceptual change. Among them, Piaget's stage theory was most prominent and has influenced virtually all trades of research in cognitive development. However, numerous theorists have seriously challenged Piagetian theories over the past decade [1, 2, 3, 4, 6]. One of the main criticisms these theorists raise against the Piaget's theory was its lack of precise specification of the mechanisms underlying conceptual change. Most recently, Siegler and his colleagues have proposed a new approach, the microgenetic analysis, to unravel the process of conceptual change [5, 6, 7]. In essence, the microgenetic method entails a dense sampling of observations so that a concept under rapid change and development can be effectively described and analyzed. In particular, [5] has suggested five dimensions or aspects to reveal the change process, namely path, rate, breadth, variability and sources of change [6]. In the present study, we adopted the microgenetic approach to examine junior-high school students' understanding of basic mechanics. Their understandings were assessed either in a group setting or individually. In the former the participants were merely required to interact with a computerized test-bank software. In the latter, the participants were required to provide explanations to their answers in addition to interacting with the test-bank software. The main reason to have such a manipulation was because there is evidence indicating that self-explanations could promote learning, especially in the conceptual domain [5, 9].

2 Method

Participants. In order to find two groups of junior-high school students with comparable competency in elementary mechanics, we first administered a paper-and-pencil conceptual test of mechanics to 280 junior-high school students from 6 classes of a private high school in the Chiayi County in Taiwan. We then selected 12 among the class of students that had a mean score close to the average performance of the entire sample. Those students had scores that were right at the level of class average. They were randomly assigned to two groups that were tested either individually or in a group setting.

Materials and apparatus. In order to effectively assess participants' conceptual development in their understanding of basic mechanics, we first built a computerized test-bank software. The test bank contained multiple-choice questions with multimedia presentation (see Figure 1 for illustration) and covered nine different units of basic mechanics, namely, (a) displacement and its magnitude, (b) average and instant velocity, (c) 2-D coordinate systems, (d) X-T graphs, (e) V-T graphs, (f) translation between X-T and V-T graphs, (g) motion equations, (h) Hook's law, (i) static equilibrium, and (j) vectors (differentiation and integration). For each unit, we first established the levels of conceptual sophistication that seemed to be appropriate for that unit. The levels represent a progression from rudimentary understanding to elaborate mastery of a given conceptual domain. Due to the variation in conceptual complexity of each unit, the levels of sophistication varied from 3 to 5, reflecting the relative difficulty and complexity among items constructed for each level. There were 10 streams of parallel items constructed for each conceptual unit; as a result the test items for each unit varied from 30 to 50 items. The items were parallel in the sense that only the protagonists and/or numerical quantities were altered between items at the same level of sophistication. Although we constructed a complete set of test bank, only units of *a* (displacement), *c* (2-D coordinate system), *d* (X-T graphs), *h* (Hook's law), and *i* (static equilibrium) were administered to the participants due to the constraints of available time and the background knowledge covered in their regular courses on mechanics.

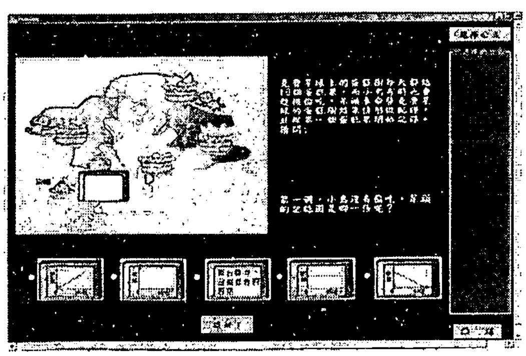


Figure 1. An example of computerized test item for assessing conceptual understanding of basic mechanics.

Procedure. The participants were tested in two groups, six in each group. For the group-test participants, they were assessed in a group setting (in the school's computer room), although their interactions with the test-bank software were essentially independent of one another. For the individual-test participants, they each interacted with the software separately via a notebook. Their interactions with the test-bank software, including the answers they gave for each item and the explanations they offered for their answers, were videotaped. The test-bank software was also equipped with a database for recording various aspects of participants' interactions with it, including the item number, the level of sophistication for a given item, the answer, the accuracy of the answer, the reversal index, and exit type, among others.

Participants' understanding of the five units in elementary mechanics was each assessed four times for both groups, over a period of about 4 months. The first two assessments were conducted toward the end of the spring semester and the second two assessments were conducted at the beginning of the fall semester, interrupted by the summer break. For each assessment, we adopted an adaptive testing principle by using the staircase method typically used in psychophysical research [8] for assessing the threshold. The staircase method we used entailed raising one level of difficulty (sophistication) after correctly answering two consecutive items at the same level, and lowering the level of difficulty whenever an incorrect answer was encountered. According to Levitt (1971), this procedure would yield a (conceptual) threshold value of about .71, a value that is normally used in psychophysical research. When participants answered incorrectly on an item, they were subsequently given items that were at a lower level. If they answered correctly on items that were presumably easier, they would be given items at a higher level of difficulty. At this juncture, a reversal point would be registered as the level of difficulty for items that were preceded *and* followed by items at a higher level of difficulty. A second type of reversal point has the opposite property, namely items that were preceded and followed by items that were at a lower level of difficulty. For each round of assessment we collected 5 reversal points before allowing the participants to exit from the test. The mean of the five reversal points was then used to define the level of conceptual understanding for the participant.

We also designed alternative routes for exiting the test bank software. Some of the units were relatively easy such that participants were able to correct throughout all levels of difficulty. If that happened, we would allow them to exit when they answered correctly three items in a row at the highest level of difficulty. In contrast, some units were relatively difficult, at least in the first round of assessment, such that participants were unable to advance themselves from the first to second level. We also allowed the participant to exit if they were incorrectly on three items consecutively at the first level.

3 Results

We first computed, for each of the five units examined, the mean value of conceptual threshold for each participant for each of the four rounds of assessment. These mean values of threshold were then submitted to a 2 (group) x 4 (round) mixed analysis of variance (ANOVA) for each unit separately. As can be seen in Figure 3, the differences between the two groups of participants did not reach significance level for four of the five units, namely, displacement, coordinate system, Hook's law, and static equilibrium, F's < 1 or p's > .15. The difference between the two groups approaches significant for the unit of X-T graphs, $F(1, 9) = 4.90$, $p = .054$, indicating that on average the individually tested participants ($M = 2.68$) performed better than their group-tested participants ($M = 2.05$) The main effect of round of assessment was highly reliable for two of the five units, $F(3, 27) = 6.38$, $p = .002$, for displacement, and $F(3, 27) = 5.49$, $p = .004$ for X-T graphs. It was marginally significant for the unit of Hook's law, $F(3, 27) = 2.84$, $p = .057$, but was unreliable for units of 2-D coordinate system and static equilibrium, F's < 1.

Because the participants were tested four rounds in succession, the data allow us to perform trend analyses in addition to the omnibus ANOVA. The results of trend analysis for the three units that participants appeared to undertake rapid change reveal the following findings: For *displacement* unit, both the linear trend and the cubic trend were reliable, $F(1, 9) = 8.51$, $p < .02$, and $F(1, 9) = 12.45$, $p < .01$, respectively. Likewise, for *X-T graphs* unit, both the linear and cubic trends were reliable, $F(1, 9) = 5.79$, $p < .05$, and $F(1, 9) = 7.25$, $p < .03$, respectively. Finally, for *Hook's law* unit, only the cubic trend was reliable, $F(1, 9) = 10.68$, $p = .01$, but the linear trend was not, $F(1, 9) = 2.25$, $p > .16$.

4 Discussion and Conclusion

The findings of the present study indicate that three of the elementary concepts in mechanics we examined—*displacement and its magnitude, X-T graphs,* and *Hook's law*--were under rapid change such that with four rounds of assessment, spanning a period of 4 moths, we had witnessed nontrivial change over time. It is interesting to note that the pattern of conceptual change for both displacement and X-T graphs units not only exhibited a pattern of monotonic increase in level of sophistication, and thus yield reliable linear trends, but also exhibited reliable cubic pattern, indicating that the conceptual understanding was not as stable as a stage theory would have predicted. That is, almost all participants, regardless of the setting in which they were tested, exhibited the pattern that while they had a better performance at the second round of assessment, their performance dropped on the third round of assessment before they advanced themselves again at the four round (see Figure 2). Those who have criticized the stage theories such as Piaget's have noted such a pattern. According to stage theories, participants should at least remain at the same stage of development once they reach at a given stage. It is in this sense that the microgenetic approach can offer a picture that perhaps is closer to the reality of developmental course. There were also units, namely, the *2-D coordinate system* and *static equilibrium*, to which our participants demonstrated their understanding and mastery early on such that no substantial change was observed over the period of assessment. These differences among the units once again demonstrates the strength of microgenetic method in that not all conceptual domains would undertake a uniform course of development. Finally we were somewhat surprised to find that self-explanations did not exert reliable effects on our participants' performance. One possible reason may have to do with the fact that the test items we constructed were really geared toward participants' basic conceptual understanding, In so doing we may have greatly reduced the complexity of those conceptual domains such that whether or not self-explanations were required was ineffective in promoting conceptual change. [9]

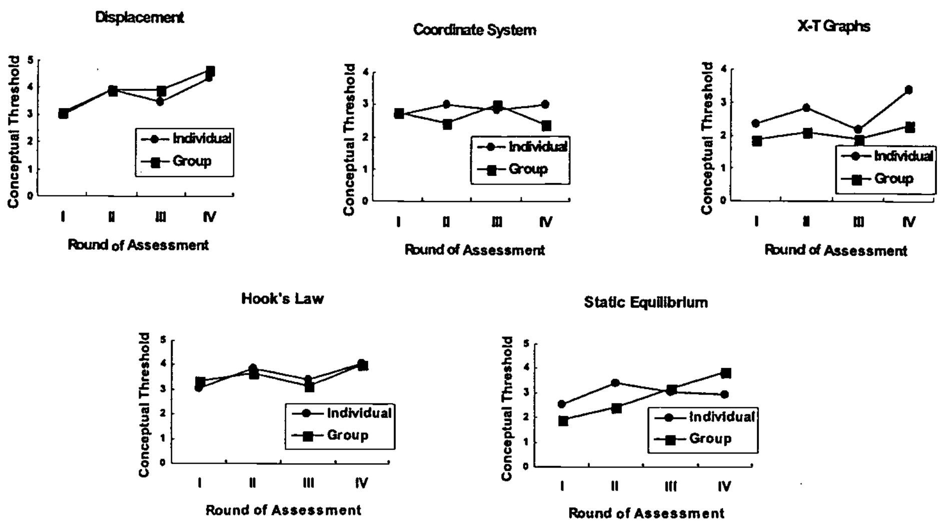


Figure 2. The conceptual threshold value for each unit as a function of round of assessment and test setting (individual vs. group).

References

[1] Case, R. (1992). Neo-Piagetian theories of intellectual development. In H. Beilin & P. B. Pufall (Eds.), *Piaget's theory: Prospects and possibilities* (pp. 61-106). Hillsdale, NJ: Erlbaum.

[2] Halford, G. S. (1993). *Children's understanding: The development of mental models*. Hillsdale, NJ: Erlbaum

[3] Klahr, D. (1995). Computational models of cognitive change: The state of the art. In T. J. Simon and G. S. Halford (Eds.), *Developing cognitive competence* (pp. 355-376). Hillsdale, NJ: Erlbaum..

[4] Klahr, D. (1999). The conceptual habitat: In what kind of system can concepts develop? In E. K. Scholinick, K. Nelson, S. A. Gelman, and P. H. Miller (Eds.), *Conceptual development: Piaget's legacy* (pp. 131-162). Hillsdale, NJ: Erlbaum.

[5] Siegler, R. S. (1995). How does change occur: A microgenetic study of number conservation. *Cognitive Psychology, 25*, 225-273.

[6] Siegler, R. S. (1996). *Emerging minds: The process of change in children's thinking*. New York: Oxford University Press.

[7] Siegler, R. S., & Crowley, K. (1991). The microgenetic method: A direct means for studying cognitive development. *American Psychologist, 46*, 606-620.

[8] Levitt, H. (1971). Transformed up-down methods in psychoacoustics. *Journal of Acoustic Society of America, 49*, 467-477.

[9] Chi, M. T. H., Bassok, M., Lewis, M., Reimann, P., & Glasser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science, 13*, 145-182.

Peer Help for Problem-Based Learning

Susan Bull and Jim Greer
ARIES Laboratory, Department of Computer Science,
University of Saskatchewan, Saskatoon, SK S7N 5A9, Canada.
bull@cs.usask.ca, greer@cs.usask.ca

This paper describes the I-Help peer help network, where helpers and helpees are paired according to the contents of their user models. Although originally designed for large groups, in this paper we suggest ways in which I-Help may be used in a small group, problem-based learning curriculum. The use of I-Help will be very different in this context: it is not expected to be necessary for all students. However, some learners may experience difficulties with some aspects of problem-based learning, such as: scheduling of meetings; involvement in discussions; understanding roles; acquiring skills for problem-based learning; different interaction preferences; differences in cognitive styles. We describe how I-Help may be used to alleviate some of these difficulties, in particular: by putting groups into contact with other groups; or putting individuals into contact with someone outside their group who can advise, or who is facing similar problems, and would like to explore the issues jointly. At the same time, group cohesion is not disrupted.

Keywords: **peer help, problem-based learning, student modelling.**

1 Introduction

Problem-based learning (PBL) is used in many academic subjects (e.g. architecture, business, education, engineering, law, medicine). The first implementations were in medical education, and PBL is still used in many medical sciences courses today. We therefore focus on medical education in this paper, though many of the arguments are applicable to a range of subjects.

Medicine is a difficult subject to teach and learn: the knowledge to be acquired and integrated is broad and very complex. This knowledge is useful only if it can be applied to problems presented by real patients. Such problems are ill-structured, specified with partial information, and often complicated by diverse interacting factors. While acquiring basic domain knowledge is a fundamental activity in medical education, integrative problem-solving is also a fundamental goal.

PBL attempts to focus learning around authentic patient problems or cases, which bring together many interacting issues of a multidisciplinary nature. A core aspect of PBL is that problems should be only partially specified. PBL involves the student in a practical activity, carried out in small groups (usually 4-8, facilitated by a tutor) in which students identify and research their own learning issues [17]. Typically a group will meet to discuss a case, identify learning issues, and then research these individually using a variety of resources (e.g. print-based, web-based and people). They then meet again to report and discuss the case further.

Investigations into the benefits of PBL have produced mixed results, possibly in part because traditional assessment mechanisms are less appropriate measures of the goals of PBL [13,30]. It is stressed that there is, as yet, no evidence that a PBL curriculum is more successful than a traditional approach [27]. Nevertheless, PBL has been embraced by some as the preferred approach to medical education, advantages cited including: the self-directed nature of PBL [27]; a greater tendency towards a deep approach to learning [21]; and positive student attitudes [6]. Others suggest that acquisition of basic domain knowledge may not be well supported in PBL. Learners may later recall less factual knowledge, since they are spending time learning other skills in addition to content [30], and they may lack depth of knowledge [18]. Explanations generated by PBL students can be less coherent, and more frequently incorrect [23]. Learners may also become bored with the PBL process [29]. It has also been recognised that PBL may simply not suit all students' ways of learning [10]. While the peer help system described in this paper can assist in a number of areas, it is this latter aspect that we focus on here.

This paper is neither a critique nor an endorsement of PBL. However, we emphasise that in PBL (as in traditional education), there is a need for tools to support peer interaction for situations where learners need assistance. In this paper we describe how the I-Help (Intelligent Help) system can be used to support students who have difficulties with the PBL approach by putting groups into contact with other groups, or an individual into contact with another learner who may advise or collaborate.

Section 2 of this paper introduces existing examples of computer support for PBL, and describes other systems which mediate peer help. The advantages of I-Help in large groups are described in Section 3. Section 4 discusses how the large group implementation of I-Help may be adapted to support PBL students when they are experiencing problems with the PBL approach. Conclusions are presented in Section 5.

2 Computer support for problem-based learning and peer help

Computer support for group interaction in PBL has been implemented for the asynchronous distance education context; the synchronous distributed learning context; and the co-present small group situation. Kamin et al. [15] describe a combined Web/CD-ROM program containing a video patient case, for use by a group of third year medical students and tutor. It is designed to facilitate asynchronous PBL during a clinical course component, requiring independent and collaborative involvement. Cameron et al. [5] discuss a distributed problem-based learning project using conferencing software together with a web page, to support synchronous sessions aimed at enabling 'authentic PBL' to occur amongst distributed first/second year medical students and a tutor. Koschmann et al. [16] introduce a method of conducting PBL meetings between students and tutor in a face-to-face context, using connected individual laptops and a large shared display. This approach is close to that found in PBL meetings not supported by computers, but offers some advantages: parallel polling (to ascertain each group member's views before they hear the ideas of others); and a record of contributions.

Computer support for PBL may, or may not include actual cases within the program: students may be collaborating about computer-presented cases, or interacting through the computer environment about externally introduced cases. External cases may be provided by the tutor off-line, or may be drawn from a database of patient cases (e.g. PATSy [19]). Systems to support PBL may help to structure and focus PBL discussions. However, even where such systems are available to a student, we believe that additional support is needed by some learners, to help them cope with the PBL situation if they feel uncomfortable with some aspects of it.

While it is acknowledged that many learners benefit from collaborative work, it is also the case that collaboration will not suit all learners; or a particular instantiation of a computational or non-computational collaborative learning environment may not suit a learner who could potentially gain much from collaborative interaction. Thus more flexible means of facilitating peer interaction would be useful. This kind of support will differ from that provided by systems such as the above: students who find the PBL approach difficult may find it useful to be put into contact with a peer who can share experiences about specific aspects of PBL.

An increasing number of peer help systems are attempting to organise learner interactions according to the student models of the individuals concerned – i.e. they have a matchmaking component; or by learner selection of available helpers. The matchmakers in such systems can take account of a variety of factors, but they most often look at students' relative proficiencies in the target domain. A few examples are given below.

An example of a peer help environment is that of Yu et al. [31], where more advanced learners act as mentors. Mentors are selected according to their knowledge, with reference to the following criteria: students who have successfully completed the course; students with high grades in other courses; students who have finished assignments; students who have successfully completed the computer-based tasks about which others need help; teachers and teaching assistants. The assumption is that the group of mentors and the student group do not overlap (though Yu et al. suggest extending the system to allow student-student help). Students select mentors based on availability (mentors may be involved in up to three help sessions); and the current problem (mentors may only help on one problem area at a time).

The above example has the advantage that learners choose to receive help when they need it, and are not forced into a collaborative context if they prefer not to participate. Further, they are guaranteed a knowledgeable helper. Nevertheless, there are drawbacks to this approach outside the setting for which it was designed. The set-up is very rigid: currently only externally acceptable (i.e. tutor-selected) individuals may be mentors. This does ensure that helpers are knowledgeable, but it does not require that they are good helpers. It also does not take account of the fact that students may benefit educationally from giving help, as well as receiving it.

Hoppe [14] proposes integrating knowledge from individual student models to support group learning – i.e. to parameterize group learning. One of the benefits is that peer helpers may be selected for help sessions: a knowledgeable helper can be partnered with a less knowledgeable student. In Hoppe's work this occurs as follows: a learner issues a help request; a menu of potential suitable helpers is offered; the learner selects their choice of helper; the selected helper receives the help request; the helper accepts or rejects the request. This approach is claimed to avoid personal conflicts, as helpers are neither assigned, nor must they interact directly with the helpee if they wish to refuse. It also allows all participants the opportunity to be helpers, as long as they know about the topic. It does not guarantee, however, that selected helpers will be proficient at helping.

Ogata et al. [22] extend this notion of peer help networks, taking into account pre-existing social networks amongst individuals, claiming that these are at least as consequential in a help context, as more official organisational structures. Ogata et al.'s approach allows users to register their proficiencies and social networks, and it also automatically traces user relationships by logging email exchanges. This provides additional information on personal networks, and also on abilities of the user: if an individual answers a question posed by a peer, the helper is assumed to be knowledgeable. These relationships are taken into account when matching potential helpers with those requesting help.

The above approaches allow peer interactions to be initiated by a learner, as required. Helpers are contacted, and may choose to take up or reject interactions. The first example [31] does not require extensive student models, but is quite restricted. The second example [14] expects student models to be in place, though overlay models are sufficient to indicate knowledge levels of individuals. The final example [22] does not require detailed models of knowledge, since it relies on social closeness and self-evaluations together with assumptions about competence based on question keywords in a help request, that has been responded to by the individual being modelled. However, what is not present in these approaches is an ability to match students according to their preferences of interaction method, or individual cognitive style, or to take into account a helper's ability to *help*. Such issues may be just as important for peer interaction to be successful.

The following section describes I-Help: an environment based on multiple user models, to match students who have help requests with potential peer helpers. I-Help aims to accommodate a broader range of characteristics that might be important when pairing learners. Suggestions of how I-Help might be usefully applied in PBL are then given in Section 4. This includes the more common face-to-face PBL context, and use alongside software to support group interaction in PBL, such as described at the beginning of this section.

3 I-Help

I-Help is the integration of several information/help sources brought together through the metaphor of a help-desk [12], designed originally for large student groups. The two principal components are an asynchronous public discussion forum [3], and a one-on-one private discussion facility which may be used synchronously or asynchronously. In the case of the private discussions, multiple distributed user models are used [20] to match students who can help each other in their learning. Each user has a personal agent which uses its owner's student model as a source of information for negotiating help sessions with other users, through their respective personal agents [28]. (Some examples of agent personas are shown in Figure 1.) The following illustrates the sequence of events for a help request. (For an example see [11]).

1. A student contacts their agent to issue a request for peer help;
2. The student's agent negotiates with the agents of other learners, to find appropriate helpers;
3. The top five user-matches are emailed that there is a help request waiting for them in I-Help;
4. To ensure maximum immediacy of response, while not duplicating effort, the first helper to accept the request starts a one-on-one discussion. Requests to other potential helpers are thereby cancelled;
5. Upon completion of discussion, each learner receives an evaluation form through which they evaluate their partner, for student modelling purposes.

The I-Help student model is composed, as stated above, in part from peer evaluations given at the end of a help session by both helper and helpee, about the knowledge of the other participant. The student model also comprises self-evaluations of knowledge level in each of the domain areas. In addition, helpees rate the utility of the help received. Social issues are also considered: learners can add users to their 'friends' list–i.e. people with whom they will preferentially interact, be they 'real friends' or people they do not know, but who have been helpful to them in the past. Students may also add individuals to their 'banned' list–people with whom they wish to have no further dealings. Much information for the student model is easily captured, since it is user-given. It is continually updated as peers evaluate help sessions once they are completed.

Also modelled are individuals' cognitive styles. The identification of cognitive style is based on Riding and Cheema's classification [26], which comprises two dimensions: wholist-analytic and verbal-imagery. The wholist-analytic dimension refers to the extent to which an individual usually *processes* information in *wholes* or separate *parts*; the verbal-imagery style relates to the degree to which an individual tends to *represent* information during thinking in a *verbal* or *image* form. In I-Help this information is provided through a front-end questionnaire. The questionnaire is very short, designed for students who may not themselves be interested in the outcome. The aim is to encourage learners to provide at least some information. While recognising that this is not ideal, partial cognitive style information is considered preferable to no information at all.

Five question types were identified, requiring different cognitive style combinations of helper and helpee:

1. *How does this fit with other things?*
   The first choice of helper for this type of question is a wholist, regardless of the cognitive style of the helpee, because wholists will tend to be better equipped to provide a broader overview.
2. *What are the details of...?*
   For this question type an analytic helper is preferred, regardless of whether this matches with the cognitive style of the helpee, because analytics tend to grasp the details of a topic more readily than wholists.
3. *Can you recommend any good materials for...?*
   The aim is to match individuals on the verbal-imagery dimension, since a verbal learner will more likely recommend materials helpful to another verbaliser, and an imager will do likewise for another imager.
4. *Miscellaneous question*
   This category covers any questions not included in the above. The default is to match all learners on the wholist-analytic dimension. If possible, learners are also matched on the verbal-imagery dimension.
5. *Questions requiring simple answers*
   No cognitive style matching is undertaken for straightforward questions requiring a simple answer, as cognitive styles are likely to have little impact here.

When submitting a help request, the learner indicates the question type from the above selection.

In addition to self and peer user-given information, learner models are updated automatically based on observations of eagerness (browsing and active posting behaviour in the public discussion forums, and amount of help given in private discussions). Furthermore, personal agents note which cognitive style matches seem most successful for different question types, and update the user model accordingly. (This also helps to overcome potential inaccuracies in the initial self-report.) Figure 1 illustrates the sources of information for the student model (open arrowheads), and the differences between private and public discussions. In the private discussions a learner interacts directly with a single peer in each dialogue, to give and receive help. Public discussions take place in forums – there is no direct interaction between two people (solid arrowheads).

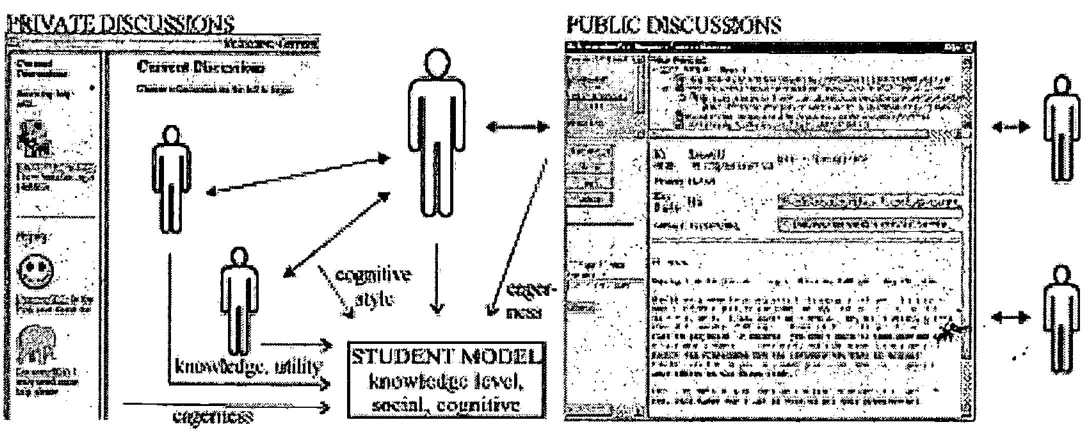


Figure 1: I-Help public and private discussions

In seeking partners, a personal agent tries to balance all relevant information (knowledge level of helpers; helpfulness of helpers; eagerness to help; preferential friends; exclusion of banned people; appropriateness of cognitive style). By default these issues are given equal weighting, but the learner may re-rank each component, as is important for them. For example, some learners may have more flexible cognitive styles. For such students, style may be a relatively unimportant factor. Other students will have more difficulty adapting to someone else's way of learning, and will assign greater importance to cognitive styles – perhaps even preferring this kind of match above the requirement that a helper should be very knowledgeable.

A variation on the peer help scenario involves permitting students to choose the kind of interaction they want, based on the S/UM system [4]. In addition to peer help, students may seek: peer feedback about work drafted or completed; collaborative learning; cooperative learning (i.e. X learns A & Y learns B, followed by tutoring or reporting). In addition to peer help, this allows students who wish to learn collaboratively or cooperatively the opportunity to find the most suitable partner. When a user sends an interaction request, they specify the kind of interaction they are seeking. Their agent negotiates a match with someone who also wishes to interact in that manner, and who has appropriate characteristics (e.g. a helper should have greater proficiency in the topic than the helpee; a collaborative partner should have a similar, non-expert, knowledge level).

In summary, the utility of I-Help increases with the number of users, as good matches become more feasible. Much of the user modelling is performed quickly and naturally by users (self- and peer-evaluations), and these models by themselves are sufficient even early during interactions, before additional system modelling has occurred. Student models contain content, cognitive and social information, which can be ranked in order of importance by learners. Further, I-Help can easily be applied across a broad set of courses: all that is required is a course description (in the form of course component labels) to be provided by the course tutor. Knowledge levels represented in user models, to contribute to matchmaking, are then related to these labels. Apart from reducing the load on tutors, from students requesting information, there are three major educational benefits:

- Students receive help when they have difficulties;
- Students learn through encountering the possibly conflicting viewpoints of others;
- Students will necessarily reflect on an issue when giving help on it.

Thus it is not only those receiving help, who benefit.

4 I-Help in problem-based learning

Due to the nature of PBL, students undertake a lot more research than traditionally educated learners, relying less on teacher-recommended texts. Many students use electronic resources more heavily than other resources [8], and they also use general library resources more extensively than their traditional counterparts [2]. I-Help provides additional human resources, forming a natural extension of this situation, and is likely to be useful to many students in PBL during the research phase. However, in this paper we focus on supporting those students who are uncomfortable with some aspects of the PBL approach itself.

Since PBL is focussed around small pre-established peer groups it is less obvious how I-Help might be applied, as opposed to in larger, traditional classes for which it was originally designed. Nevertheless, as illustrated in the following description, there are a number of situations in which I-Help could be useful in PBL.

There are a variety of potential difficulties to take into account in a PBL course. For example:

- It can be difficult for some students to find time to meet outside scheduled class hours;
- For a group to function effectively, individual team members should all be involved in group discussions;
- Students may not fully understand their role in the group;
- Students may lack the skills to make group interactions work;
- Students have different interaction preferences;
- Students have different cognitive styles.

Considering the first two of these issues, the public discussion forum of I-Help is a useful tool to keep all students in contact with their own group's discussions, but also allowing interaction between particular group members, should help or clarification be needed by some participants, on some group issue. At the same time, all students remain up-to-date with all interactions, at a time that suits them, thus freeing up part of meeting times for questions and group issues less easily handled through computer interaction.

Perhaps more unusual in the PBL context: there may be occasions when students could usefully interact *across* groups. As suggested above, it is not the aim to exclude any group members from any interaction important for group progress, but there may be situations where individuals from different groups could help each other, on issues perceived as not directly relevant to either group as a whole. For example, in some PBL contexts the various roles are divided amongst group members, often rotating. In such situations it might be helpful if individuals from different groups who are performing the same role (e.g. scribe; group leader; information analyst), could interact – especially if it is early in the rotation, and there is less group experience on which to draw. It will also be helpful for students finding their role difficult, who are part of a group whose members do not appreciate the learner's problems. Their personal agent could locate a helper who has successfully fulfilled the responsibilities of the role in the past, or find another student with similar problems, with whom they can jointly explore aspects of the role. Where the whole group acknowledges a lack of understanding of any role, one of the group members may seek outside assistance on behalf of everyone.

I-Help's user models must therefore be extended to include information about student roles. I-Help must know the current role of individuals, in order to put students in touch with others facing the same tasks; and it should also remember the roles that individuals have previously held, and whether they were competent, and whether they are willing to offer help to novices in these roles. I-Help may then be used to pair individuals in interactions relating to role responsibilities, keeping such interactions amongst those for whom the discussion is currently relevant and/or helpful. As more students come to perform each role, previous help session archives may be accessed as help resources. In this manner, it is hoped that more students may develop an understanding of how to meet their various responsibilities, resulting in improved group functioning.

It has been commonly noted that many students entering the medical sciences do not possess the skills necessary for effective group interaction in PBL – e.g. discussion, decision making, conflict management, leadership, group feedback processes [24]. Although I-Help does not teach these skills, its ability to match students with others who are in a similar position, or who are able to help, provides a form of support not usually available. If a single individual has problems, the other group members may be able to compensate while also supporting the learner's development of the skill concerned. However, where group members recognise a general deficiency, they may use I-Help to put them in contact with a group that functions effectively with regard to the particular skill. They may be invited to observe, as the efficient group models the behaviour during their next meeting, or one of the effective group members may describe how their group tackles these issues. This will be especially useful where there are no resources (e.g. time, staff) for skills training.

Again the I-Help student model must be extended, to accommodate information about group interaction skills. This will involve all groups in a group evaluation process in order that they may provide skills information for the user model, which in itself will be a useful reflective activity. The main difference in the structuring of the model in this case is that skills information will relate to *group* functioning, and not to specific group members. Thus I-Help must also know which students belong to which groups. Skills information need then only be given by one learner.

A potential difficulty encountered by a student who might otherwise do well in PBL is that other group members may have different interaction preferences: some students gain much from brainstorming or spontaneous discussion, while others prefer to reflect and organise their thoughts before communicating. The combination of face-to-face meetings and the public discussion forums helps to cater for all students, while the possibility also exists to arrange collaboration, cooperation and feedback through the private discussions.

Students also have different cognitive styles. Some individuals understand verbal descriptions well, while others need pictures, diagrams, or demonstrations. Some learners deal well with abstract concepts and detail, while others tend towards a more general overview. Although a mixture of cognitive styles might sometimes be complementary in a group setting, and have a positive effect on group performance, some cognitive style combinations may lead to difficulties for some individuals. For example, if most members of a group are analytic, a wholist learner may have difficulty gaining the overview perspective they require to integrate information. Such an individual might find the situation very difficult as a learning experience. It is also possible that the other group members will not understand their difficulty. This is a problematic situation since all group members should be involved in group communications for a group to feel comfortable and function well. Full participation is essential in some groups to avoid resentment by other group members if they feel that one person is not contributing. I-Help private discussions should not, then, be used as an alternative to group interaction, as the group may suffer as a result. However, for students who have problems adapting to the way the other group members work, I-Help may provide a much-needed 'lifeline' by matching them with a student with a similar cognitive style, to support their PBL activities in a 'more comfortable' fashion. Thus they will continue to interact with their group to the best of their ability given the difficulties they experience, but they may also work with another learner outside the group context if they feel this to be useful. This need not detract from the group experience as a whole, since the learner may report back any findings. Taking the above example, such an individual's contribution may now be greater, since they will be able to provide the overview that the analytics lack. Therefore their group contribution may be stronger than any earlier contributions where they had not had this additional learning opportunity, and were interacting only within the confines of the particular group's interaction dynamics.

This section has suggested a number of ways in which I-Help might be useful in PBL. It is not suggested that all PBL students should use it (though the public forum is likely to be generally useful), but that I-Help could arrange peer support in cases where an individual is having difficulties with some aspect of the PBL approach.

Although it does not address the problem of group learning for an individual who prefers to learn alone, or in a different kind of group situation, it does at least provide them with some support that they would otherwise not have.

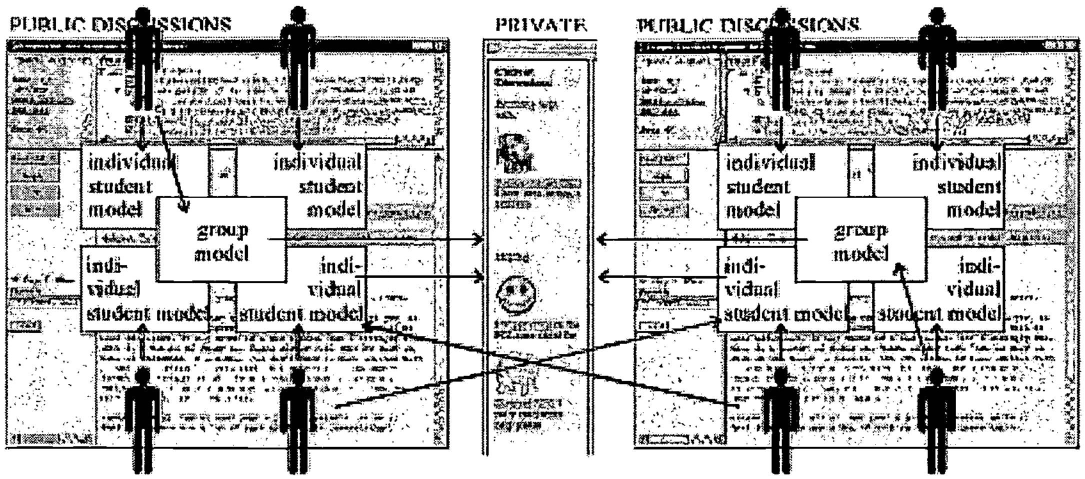


Figure 2: I-Help in problem-based learning

To introduce I-Help to the PBL setting, some additions to the user models are necessary. However, these are very easy to implement, having simplicity in common with the present representations. Currently I-Help user models contain: a quantitative measure of knowledge levels in the various domain areas; a quantitative indication of helpfulness; a quantitative measure of eagerness; a list of friends; a list of banned people; identification of cognitive style; a list of preferred interaction types. The additional information proposed above comprises: a list of roles successfully performed previously (to be added by the individual); the current role of the student (also added by the individual); a list of group membership (provided by one group member); a list of group skills (based on group evaluation, the result of which is entered by one group member). Thus minimal extensions could provide essential support to learners having difficulties in PBL. Provision of this information by students should also encourage them to think about factors that help to make group interaction successful.

Figure 2 illustrates how I-Help can support learners in a PBL setting. Students and peers provide student model information as occurs in large group uses. I-Help also performs some user modelling as described previously. The main difference with I-Help in PBL is that interactions for each group are focussed primarily around public discussions, with each person communicating with other members of their own group. There is less use of the private discussions. Where private discussions do occur, matching takes place according to the student models of individuals in the manner described in section 3. In addition to individual models, in PBL group models are required in order that groups may also be brought together where difficulties are recognised by the group as a whole. Information for the group model is obtained from one of the group members.

5 Conclusions

I-Help was initially designed to promote peer help amongst a group of learners in a large class situation. Some minor extensions to the system were suggested, to enable it to be effective also for students in PBL. Despite many successes claimed for this kind of collaborative interaction, not all students will function at their best with this type of curriculum. In this paper we focussed on PBL in medical education, but the arguments should be equally applicable to other academic disciplines and small group contexts, as long as the overall student numbers are large enough to enable sufficient choices of appropriate partners for cross-group interaction.

In addition to large and small group formal educational settings, I-Help might also be used beyond the classroom to support medical practitioners. For example, while some contexts have adequate funds to implement elaborate means of telemedicine (e.g. the U.S. Army [1]), remote areas which might benefit from access to various forms of telemedicine often find that the low population density does not provide sufficient demand to justify the expenditure required [25]. In rural locations a system like I-Help would provide a low cost means of obtaining expert help at least for some cases. Furthermore, practitioners requesting assistance do not themselves need to know who is the best person to contact. Similarly, I-Help might be useful in putting into contact physicians who would like to hear experiences of other practitioners. For example, where ethical considerations are important to a case, such as conflicts between medical advice and parental beliefs [7]. I-Help might also be used alongside diagnostic decision support systems in cases where physicians remain unsure about hypotheses, since the advice offered by such systems may sometimes be misleading [9]. Experience with I-Help at university should encourage more individuals to register once they graduate and specialise.

Acknowledgements

This work is funded by the Canadian Telelearning Network of Centers of Excellence, Project 6.28. We thank the ARIES Lab members, in particular: G. McCalla, J. Vassileva, R. Deters, J. Cooke, L. Kettel and J. Bowes.

References

[1] Berry, F.C. "Telemedicine and the army", ARMY Magazine, April 1996.
[2] Block, K.J. "Problem-Based Learning in Medical Education: Issues for Health Sciences Libraries and Librarians", Katharine Sharp Review 5, 1997.
[3] Bowes, J. "Knowledge Management through Data Mining in Discussion Forums", MSc Thesis, Department of Computer Science, University of Saskatchewan, 2000.
[4] Bull, S. "A Multiple Student and User Modelling System for Peer Interaction", in R. Schäfer & M. Bauer (eds) 5 GI-Workshop, Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen, Universität des Saarlandes, 61-7, 1997. On-line proceedings: http://w5.cs.uni-sb.de/~abis97/proceedings/
[5] Cameron, T., Barrows, H.S. & Crooks, S.M. "Distributed Problem-Based Learning at Southern Illinois University School of Medicine", in C. Hoadley & J. Roschelle (eds), Computer Support for Collaborative Learning, Stanford University/unext.com, 86-93, 1999.
[6] Camp, G. "Problem-Based Learning: A Paradigm Shift or a Passing Fad?", Medical Education Online 1(2), http://www.med-ed-online.org/ , 1996.
[7] Coward, H. & Hartrick, G. "Perspectives on Health and Cultural Pluralism: Ethics in Health Care Education", Workshop on Cognitive and Social Science Foundation for Medical Education and Training in the Information Age, Dunsmuir Lodge, Victoria, 2000.
[8] Deretchin, L.F., Yeoman, L.C. & Seidel, C.L. "Student Information Resource Utilization in Problem-Based Learning", Medical Education Online 4(7), http://www.med-ed-online.org/ , 1998.
[9] Friedman, C.P., Elstein, A.S., Wolf, F.M., Murphy, G.C., Franz, T.M., Heckerling, P.S., Fine, P.L., Miller, T.M. & Abraham, V. "Enhancement of Clinicians' Diagnostic Reasoning by Computer-Based Consultation", Journal of American Medical Association 282(19), 1851-1856, 1999.
[10] Gillespie, R.S. "Problem-Based Learning Redefines Medical Education", IMpact (Summer), American College of Physicians, 1995.
[11] Greer, J. & Bull, S. "Computer Support for Collaboration in Medical Education", Clinical and Investigative Medicine, in press.
[12] Greer, J., McCalla, G. Cooke, J., Collins, J., Kumar, V., Bishop, A. & Vassileva, J. "The Intelligent Helpdesk: Supporting Peer Help in a University Course", in B.P. Goettl, H.M. Halff, C.L. Redfield & V.J. Shute (eds), Intelligent Tutoring Systems, Springer-Verlag, Berlin Heidelberg, 494-503, 1998.
[13] Hmelo, C.E., Gotterer, G.S. & Bransford, J.D. "A Theory-Driven Approach to Assessing the Cognitive Effects of PBL", Instructional Science 25, 387-408, 1997.
[14] Hoppe, H.U. "The Use of Multiple Student Modeling to Parameterize Group Learning", in J. Greer (ed), Proceedings of World Conference on Artificial Intelligence in Education, AACE, Charlottesville VA, 1995.
[15] Kamin, C.S., Deterding, R.D., Wilson, B., Armacost, M. & Breedon, T. "The Development of a Collaborative Distance Learning Program to Facilitate Pediatric Problem-Based Learning", Medical Education Online 4(2), http://www.med-ed-online.org/ , 1999.
[16] Koschmann, T.D., Kelson, A.C., Feltovich, P.J. & Barrows, H.S. "Computer-Supported Problem-Based Learning", in T.D. Koschmann (ed) CSCL: Theory and Practice of an Emerging Paradigm, Lawrence Erlbaum Associates, Hillsdale, NJ, 1996.
[17] Koschmann, T. & Stahl, G. "Learning Issues in Problem-Based Learning", Panel at CSCW'98, 1998. http://www.cs.colorado.edu/~public.conferences/1998/cscw98/cscw_workshop.html
[18] Lee, R.M.K.W. & Kwan, C-Y. "The Use of Problem-Based Learning in Medical Education", http://www-fhs.mcmaster.ca/mdprog/pblart.htm , 1999.
[19] Lum, C., Cox, R., Kilgour, J., Morris, J. & Tobin, R. "PATSy: A Multimedia Distributed Web-Based Resource for Aphasiologists in Research and Education", Aphasiology 13(7), 573-579, 1999.

[20] McCalla, G., Vassileva, J., Greer, J. & Bull, S. "Active Learner Modelling", in G. Gauthier, C. Frasson & K. VanLehn (eds), Intelligent Tutoring Systems, Springer-Verlag, Berlin Heidelberg, 53-62, 2000.
[21] Newble, D.I. & Hejka, E.J. "Approaches to Learning of Medical Students and Practising Physicians: some empirical evidence and its implications for medical education", Educational Psychology 11(3-4), 1991.
[22] Ogata, H., Sueda, T., Furugori, N. & Yano, Y. "Augmenting Collaboration Beyond Classrooms through Online Social Networks", in G. Cumming, T. Okamoto & L. Gomez (eds), Advanced Research in Computers and Communications in Education, ICCE '99, IOS Press, Amsterdam, 277-284, 1999.
[23] Patel, V.L. & Groen, G.J. "Reasoning and Instruction in Medical Curricula", Cognition and Instruction 10(4), 335-378, 1993.
[24] Peterson, M. "Skills to Enhance Problem-Based Learning", Medical Education Online 2(3), http://www.med-ed-online.org/ , 1997.
[25] Puskin, D.S., Mintzer, C.L. & Wasem, C. "Telemedicine: Building Rural Systems for Today and Tomorrow", in P.F. Brennan, S.J. Schneider & E. Tornquist (eds), Information Networks for Community Health, Springer, New York, 1997.
[26] Riding, R. & Cheema, I. "Cognitive Styles–an Overview and Integration", Educational Psychology 11(3-4), 193-215, 1991.
[27] Spencer, J.A. & Jordan, R.K. "Learner Centred Approaches in Medical Education", British Medical Journal 318, 1280-1283, 1999.
[28] Vassileva, J., Greer, J., McCalla, G., Deters, R., Zapata, D., Mudgal, C. & Grant, S. "A Multi-Agent Design of a Peer-Help Environment", in S.P. Lajoie & M. Vivet (eds), Artificial Intelligence in Education, IOS Press, Amsterdam, 38-45, 1999.
[29] Wilson, B.G. & Cole, P. "Cognitive Teaching Models", in D.H. Jonassen (ed), Handbook of Research in Instructional Technology, Scholastic Press, New York, 1996.
[30] Woods, D.R. "Problem-Based Learning: Helping your Students Gain the Most from PBL", McMaster University, 1996.
[31] Yu, Y-T., Shan, H-L., Chou, C-Y. & Chan, T-W. "WWW-Based Mentoring System", in Z. Halim, T. Ottman & Z. Razak (eds), Proceedings of ICCE, AACE, 381-389, 1997.

The Research on Difficulty of Asynchronous Learning Materials Based on Studying Time Distribution

Wu-Yuin Hwang and Rueng-Lueng Shiu
Information and Computer Education
National Kouhsiung Normal Univeristy
wyhwang@nknucc.nknu.edu.tw

The purpose of asynchronous distance learning systems is to enhance students' learning performance in the internet. In this paper, we investigate the characteristics of the asynchronous materials and propose the criteria to evaluate them. Employing the criteria, the materials could be adjusted to meet most students' learning pace. First, the TDC (time-distributed curve) which is a learning curve is derived from students' studying time distribution. By the TDC, it is obtained that the more difficult the materials of the chapter are the steeper the TDC becomes. Also the total learning time of each chapter indicates the quantity of the matter. Employing the total time of each chapter, we could evaluate whether the quantity of the matter is sufficient to match students' learning desire.

Keywords: **distance learning· learning portfolio· learning behavior ·learing time distribution**

1 Introcduction

1.1 The distribution of learning time with learning attitude

Teachers could interact with their students immediately at the classroom. Thus, they could get the learning behavior of their students by students' response. The learning behavior is regarded as a good measure to evaluate learning performance. But it is really hard to obtain every student's learning process and attitude because there are at least 30 students in each class. However, employing the database technology in asynchronous learning systems, it is possible to obtain all of the student's learning process and studying time.

1.2 Learning time distribution

In traditional education, students learning together in the classroom at the fixed time, and teachers control the course proceeding. But it is difficult to pay attention to all students. However, asynchronous learning systems not only provide a brand-new perspective to long-life learning but also keep track of learning time of all students. In accordance with the learning time of all students, teachers could modify the matter to match learning goals.

2 Experiment and analysis

The experimental course in our asynchronous learning system is "Basic computer concept", the materials of the course are divided into 12 chapters. The progress-control mechanism is that students need to finish the homework of the chapter in order to be promoted to the next chapter. Thirty participants engage in this experiment and they are all teachers.

The student' s learning time and login time are recorded by technologies of ASP(Active Server Page) and Database. Therefore, we could get which chapter students read and how long they read the chapter. The important curve, Time-Distributed Curve (TDC), is generated by linear regression analysis. From the slope and the area of TDC, some characteristics and results are obtained.

2.1 TDC and DCA (Degree of Course Acceptance)

Student' s reading time each chapter is recorded in our experiment. The recorded time begins from the date when the teaching materials are put in the internet for 15 days. In each chapter, all of the student' s learning time everyday is summed up.

Employing the recorded data and derived chart, each chapter has a unique TDC (time-distributed curve) by linear regression analysis. According to the time-distributed curve, teachers may decide whether the materials should be improved.

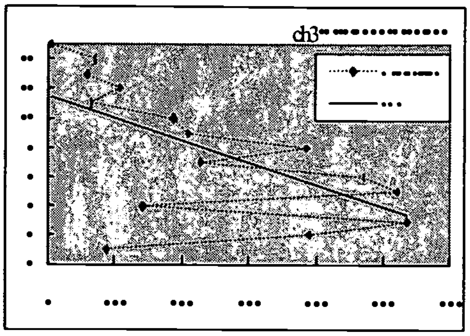


Fig.1 The TDC of ch3

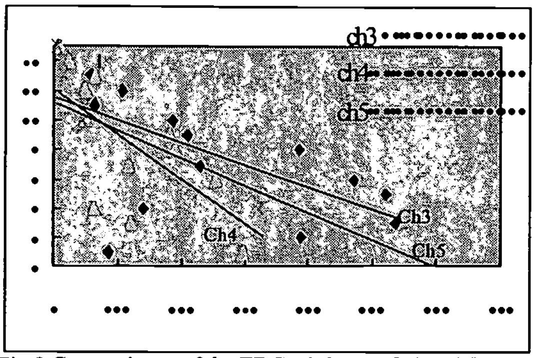


Fig.2 Comparisons of the TDC of chapter 3,4 and 5.

In Fig. 1, the X axle indicates time value and its time unit is one minute not an hour and The Y axle indicates days. For example, the total time on the 4th day is approximate 150 minutes. The slope of the TDC is minus because the total studying time would decrease while students proceed to study the matter.

The value of the slope is required to be concerned. The larger the value of the slope is, the smoother the TDC becomes. For example, figure 2 made comparisons of the TDC of chapter 3,4 and 5. Obviously, the TDC of chapter 4 has the smallest slope because it is the steepest one. And the TDC slope of chapter 3 is slightly larger than that of chapter 5. Thus, it is the most difficult to read chapter 4 and it is the easiest to read chapter 3. The reasons why the materials are hard to study may be either the materials are complicated or the user interface is not friendly to read. According to the above description, the slope of TDC could be termed as Degree of Course Acceptance (DCA, It means the harder the topic to read the smaller is the DCA.)• Besides the TDC' s slope is proposed to determine the degree of materials acceptance, there is another important characteristic, the area of the TDC, to influence the amount of learning time.

Based on the area and slope of TDC, the difficulty and quantity of the materials could be evaluated. According to the above description, it is shown that the quantity of materials would affect the amount of learner' s studying time, also the difficulty of materials would affect the length of learning period. Due to these reasons, there are two margin lines, quantity and difficulty, in Fig. 3. The two margin lines are termed e are "Margin Line Of Quantity (MLOQ)" and "Margin Line Of Difficulty (MLOD)". There are plentiful materials on the right of MLOQ, but there are poor on the left side. The upper of MLOD the materials are located the harder they are read, but lower are easy.

Since the features of MLOQ,MLOD,DCA and the area of TDC are proposed, there are four kinds of situations that the TDC represents as follow:
1. It is easy to read the material, and the contents are plentiful.
2. It is easy to read the material, but the contents are poor.
3. It is hard to read the material, but the contents are poor.
4. It is hard to read the material, and the contents are plentiful.

The MLOQ and MLOD could be employed to enhance discriminating the difficulty of the materials if the DCA and the TDC's area of the chapters are different. Finally, how is the value of the MLOQ and MLOD obtained? The MLOQ is the average of all students' learning time of one chapter. The MLOD is the average of all students' learning days of one chapter.

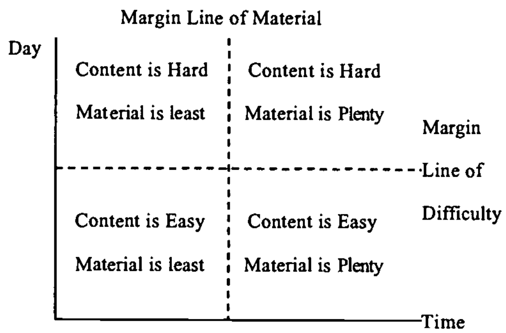


Fig.3 MLOQ and MLOD

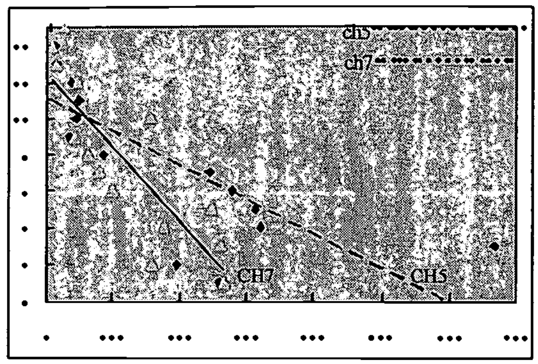


Fig.4 Compares the TDCs of the ch5 & 7

2.2 Time distribution of the interdependent course

What else may affect one's effort in the course? There are relationships between two topics. For example, there are relationships of dependency between chapter 5(Internet I) and 7(Internet II). Generally, the topic "Internet I" is dedicated to construct the fundamental concept and "Internet II" introduces the advanced ideas and practice. According to the normal teaching policy in both topics, the "Internet I" should have fewer and simpler materials than the "Internet II". Thus learners spent much less time to study "Internet I" than "Internet II".

Fig.4 compares the TDCs of the two chapters. As shown in Fig. 4, it is easy to find out chapter 7 has a smaller DCA (the slope of TDC), that is, chapter 7 is harder than chapter 5. Furthermore, the area of chapter 7 is less than that of chapter 5. The TDC of chapter 5 is located at approximately 11 on Y axle and 600 on X axle and the TDC of chapter 7 located at 12 on Y axle and 280 on X axle. According to MLOQ and MLOD as shown in fig.3, we concluded that "The chapter 7 is more difficult than chapter 5, but its quantities are much less". It is different from we described before, "Internet I" should have fewer matters than "Internet II". In our experiment, we provided much more contents in chapter 5 than chapter 7. Therefore the amount of materials in chapter 5 should be reduced.

3 Conclusions

The asynchronous learning service is an on-line collection of hypertext that provides us a new way to learn. Their students with different native intelligence come from any place and go to learn when they would like. It is very important to design and evaluate the asynchronous teaching matters so as to match teaching goals. This paper proposed some basic criteria to investigate the characteristics of teaching matters, then gave an advise to modify them to meet the learning desire. The basic criteria, the area and slope of TDC, are derived from learning time distribution. Through the basic criteria, instructors could modify the materials in accordance with most students' learning pace and talent. Especially, our proposed mechanism is worth much attention to develop the adaptive learning system. Once the asynchronous learner's studying portfolio is available, the materials could be real-time adjusted to match the learner's state.

Reference

[1] • • • • • • • • • • • • • v• • • • • • • • • • v• • • • • • • • • ;pp 13-25, 67• ;1998• •

[2] • • • • • • • • v• • • • • • • • • • • • • • • • • v, • • • • • • • • • • • • • • • • • • •

[3] • • • • • • • • • v• • • • • • • • • WEB • • • • • • • • • • • • • v• The 8th International Conference on Computer-Assisted Instruction• pp391-399,1998

Using Decision Networks for Adaptive Tutoring

Peng-Kiat Pek* and Kim-Leng Poh**
** Singapore Polytechnic*
500 Dover Road, Singapore 139651
E-mail: baiychen@magix.com.sg
*** National University of Singapore*
10 Kent Ridge Crescent, Singapore 119260
E-mail: isepohkl@nus.edu.sg

This paper reports a research project that uses dynamic decision networks in providing teacher with information on students' misconceptions and students with online tutoring. A set of Bayesian networks models the conditional dependencies between learning objectives and goals which are associated with the curriculum. Student's responses to test items are recorded and transformed as evidence into a relevant Bayesian network to compute his likely state of knowledge mastery. The personalized Bayesian network is then converted into a dynamic decision network by adding utility and decision nodes. Tutoring policy is followed through and necessary responses from the student are solicited using additional test items. The student Bayesian network is updated when new evidence arrives, and is again converted to a decision network to determine the next tutoring policy. This process is repeated until the pre-requisites are achieved. The results generated by the system and future directions are discussed.

Keywords: **Adaptive Tutoring, Decision Network, Student Model, Tutoring Strategy**

1 Introduction

Tutoring of students is an ill-structured problem that is characterized by:
(a) Uncertainty of student's knowledge mastery.
(b) Preferences, judgements, intuition, and experience of teacher.
(c) Criteria for decisions are occasionally in conflict, and highly dependent on the teacher's perception.
(d) Decisions must be achieved in limited time.
(e) The student's mental states evolve rapidly.

This study attempts to address these issues by using an intelligent *decision-theoretic* approach. The framework of this research has contributed to the development of an intelligent decision support system called *iTutor*, for tutoring Engineering Mechanics at Singapore Polytechnic.

Probabilistic or Bayesian networks [9] and decision analysis [5] have shown to be capable of solving many real-world problems involving reasoning and decision marketing under uncertainty. Bayes's nets allow for efficient reasoning and inference about combination of uncertain evidence. Student modeling with Bayes's nets for intelligent tutoring had achieved successes, see for example in [16], [11], and [2]. The differences in these works lie mainly in the choice of variables and granularity of the models.

In Villano's Knowledge Space Theory, the basic unit of knowledge is an *item* (in the form of a question). The student's knowledge state is defined as the collection of items that the student is capable of answering. The collection of all feasible states is called the *knowledge structure*, and it is connected by the *learning path*. By incorporating uncertainty at each node, the knowledge space can be transformed into a Bayes's net. The Bayes's net then constitutes a student model where probabilistic reasoning can be performed when evidence is available. Reye on the other hand, uses pre-requisite relationship of domain knowledge and *dynamic belief network* for modeling student's mastery of a topic. Finally, Conati and Vanlehn make use of teacher's solution(s) as the ideal model to track student' s faulty knowledge as the student solves a problem.

Our work here differs from others in that we construct relevant Bayes's nets by modeling learning objectives (L), evidence (V) from student responses, application of knowledge to different situations (C), and learning goal (G). A decision network [3] is then formed by adding decision and utility nodes to the Bayes's net. As it is computationally intractable to track student' s solution in real time, we use sequential decisions to generate tutoring strategy that anticipates students' responses.

This paper is organized as follows: Section 2 provides an overview of the conceptual framework for the decision theoretic intelligent tutoring system called *iTutor*. The transformation of student's responses to evidence is discussed in Section 3. Section 4 illustrates how the student model is constructed from a set of Bayes's nets, while Section 5 presents the tutoring strategy model using two-step look-ahead decision network. The results of a typical *iTutor* session are illustrated in Section 6. It emphasizes the automation of decision network construction and shows that when student's responses are available, the system is able to diagnose student's misconceptions and to provide adaptive tutoring using the generated strategy. Finally, we conclude by discussing future directions.

2 Framework of Adaptive Tutoring

Figure 1 shows the essential components of adaptive tutoring in *iTutor*.

The ***Evidence Model*** converts the student response (x_{ijk}) to item i into evidence of knowledge mastery for a relevant learning objective (v_{jk}).

The ***Student Model*** consists of a set of Bayes's nets with nodes that are either Evidence, Case, Learning Objective, or Goal. These nodes are initialized with prior information from the teacher' s judgement and theoretical probability models. The student model can be subsequently updated to reflect a student' s knowledge mastery when evidence is available.

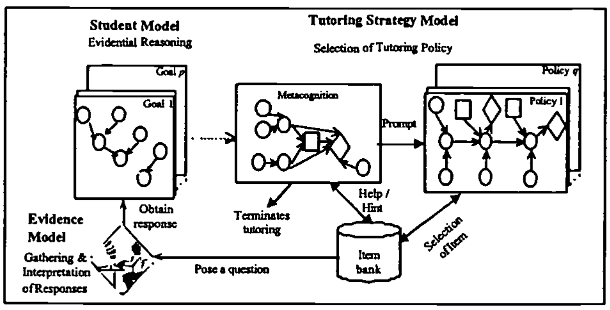


Figure 1: Inferencing Kernel of Adaptive Tutoring

The ***Tutoring Strategy Model*** uses decision-theoretic approach to select *satisfying* [14] learning objectives for tutoring student. The metacognition sub-module determines the appropriate tutor' s action: providing more help or hint, prompting another question, or stop the tutoring session. Dynamic Decision Network (DDN) provides approximate solutions for partially observable Markov decision problems, where the degree of approximation depends on the amount of look-ahead. If the decision is to obtain evidence of mastery on a learning objective, an item of difficulty b_i that matches the student' s ability θ will be selected. Student's response is collected, evaluated, and transformed into evidence at the relevant nodes in the student model. The *chance* nodes in DDN are updated and a decision policy is generated. In this way, the system is able to adapt tutoring to the needs of the student and achieve the objectives of the curriculum.

3 Evidence Model

The student' s responses are processed in the evidence model. Let V_{jk} be the evidence node that indicates the student' s (j) mastery state of learning objective k. Let X be the set of responses and $x_{ijk} \in X_{ik} \subseteq X$ be the response to item i which tests the k^{th} learning objective, then

$$\Pr(v_{jk} \mid x_{ijk}) \propto \Pr(v_{jk}) \prod_i \Pr(x_{ijk} \mid v_{jk})$$

where $\Pr(v_{jk})$ is the prior probability which can be obtained statistically from past data. $\Pr(x_{ijk} \mid v_{jk})$ is the likelihood of correct-answer score. An example of the likelihood function is $\delta_k \exp(-b_i v_{jk})$ where δ_k is the importance of knowing learning objective k so as to answer item i correctly and b_i is the difficulty index for item i.

4 The Student Model

The Student Model consists of a set of Bayes's nets, and each Bayes's net models the student' s mastery of a key concept (*goal*). In Section 4.1, the structure of the student model is defined. The construction of Bayes's net and the conditional probability assignment are discussed in Section 4.2. Instantiation of an evidence node activates a message passing process in the Bayes's net. This process results in the updating of marginal probabilities at the nodes. Most commercial software for developing probabilistic network possesses efficient algorithm [1] for implementing the message passing process.

4.1 Semantics of the Student Model

The Student Model is a directed acyclic graph (DAG) that represents a joint probability distribution of a key concept and several learning objectives. A node represents the learning objective as a random variable, and an arc represents possible probabilistic relevance or dependency between the variables. When there is no arc linking two nodes, it indicates probabilistic independence between the variables. In this study, the variables are classified into four types Evidence, Case, Learning Objective, and Goal as shown in Figure 2.

More formally, a student model in *iTutor* is a DAG $\Im = (N, \psi)$ where $N = N_v \cup N_L \cup N_C \cup N_G$ are the nodes such that N_v is a set of evidence nodes, N_L is a set of learning objective nodes, N_C is a set of case nodes, and N_G is a set of goal nodes.

Symbol	Name	Function
N_v	Evidence Node	It contains knowledge states based on student' s response.
N_C	Case Node	It contains knowledge states that reflect ability to apply knowledge in different situations (cases).
N_L	Learning Objective Node	It contains knowledge states of key learning objectives (defined in the syllabus).
N_G	Goal Node	The concept student is expected to know. Each Bayes's net must have at least one Goal node.

Figure 2: Types of Nodes in Student Model

$\psi = \psi_{pL} \cup \psi_{pC} \cup \psi_{pG}$ are the arcs such that $\psi_{pL} \subseteq N \times N_L$ are arcs into learning objective nodes, $\psi_{pC} \subseteq N_v \times N_C$ are arcs from evidence nodes to case nodes, and $\psi_{pG} \subseteq (N_L \cup N_G) \times N_G$ are arcs from learning objective or goal nodes to the goal nodes.

Notice that evidence nodes have no parent node and only evidence nodes could be the parents of case nodes. Goal nodes are always sink nodes and they have parents that are either learning objective nodes or goal nodes. This signifies that mastery of a concept (goal node) is dependent on the mastery of learning objective(s) and/or pre-requisites (other goal nodes).

4.2 Construction of a Bayes's Net

Figure 3 shows a Bayes's net on mastery of a hypothetical concept (*goal*) "XYZ". Each node has three knowledge states: *non-mastery, partial-mastery*, and *mastery*. The granularity of Bayes's net depends on the number of nodes and its states. However, as the granularity becomes finer, the number of entries in the conditional probability table grows exponentially.

Values at the root nodes are known as *prior* probabilities while that at other nodes are *conditional* probabilities. To use the probabilistic network the random variables must be initialized with prior probability values. These values may be based on teacher's belief or past statistics. An intuitive method is to generate a probability table based on seven-category of the difficulty of learning objectives (see Table 1). These probability values are to be input as the prior probability of the related evidence. The teacher also has the flexibility to amend the values based on their belief and context of usage. On the other hand, the probability values can be obtained from statistics of previous tests/examinations. A simple procedure for the use of past statistics is:

a) Assigned learning objectives to each question;

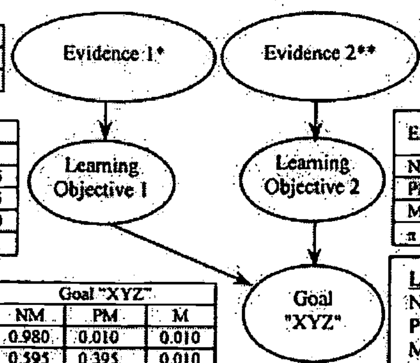
Table 1 Category of Difficulty in Mastering the Learning Objective

	Probability values		
Category	NM	PM	M
very easy	0.001	0.009	0.99
easy	0.01	0.09	0.90
fairly easy	0.05	0.15	0.80
Neutral	0.10	0.20	0.70
fairly difficult	0.20	0.30	0.50
difficult	0.30	0.40	0.30
very difficult	0.40	0.50	0.10

b) Enter student's responses (in terms of percentage) for the questions that she has answered;

c) Compute the average number of students (in percentage) for each mastery category: 040 (non-masterystate), 40-70 (partial-mastery state), and 70-100 (mastery state).

NM	0.30
PM	0.40
M	0.30



NM	0.30
PM	0.50
M	0.20

E1	L1		
	NM	PM	M
NM	0.990	0.005	0.005
PM	0.005	0.990	0.005
M	0.005	0.005	0.990
π	0.301	0.398	0.301

E2	L2		
	NM	PM	M
NM	0.990	0.005	0.005
PM	0.005	0.990	0.005
M	0.005	0.005	0.990
π	0.301	0.497	0.202

L1	L2	Goal "XYZ"		
		NM	PM	M
NM	NM	0.980	0.010	0.010
NM	PM	0.595	0.395	0.010
NM	M	0.595	0.010	0.395
PM	NM	0.395	0.595	0.010
PM	PM	0.010	0.980	0.01
PM	M	0.010	0.595	0.395
M	NM	0.395	0.010	0.595
M	PM	0.010	0.395	0.595
M	M	0.010	0.010	0.980
π		0.301	0.435	0.264

Legend
NM: non-mastery
PM: partial-mastery
M : mastery
E1: Evidence 1 contributing to mastery of L1
E2 : Evidence 2 contributing to mastery of L2
L1 : Learning Objective 1
L2 : Learning Objective 2
π : Marginal Probabilities

Note: * The prior probability for Evidence 1 is obtained from Table 1 (difficult Learning Objective 1)
** The prior probability for Evidence 2 is obtained from statistics

If a probability distribution function is able to describe the statistics, it can be used. In Figure 3, the values Pr(E2=non-mastery) = 0.30, Pr(E2=partial-mastery) = 0.50, and Pr(E2=mastery) = 0.20 are obtained from statistical data for this particular evidence. It is acceptable for another person to assign different probability values so long as it is consistent with the probability axioms [12]. Since the decision theory approach is normative rather than descriptive, it is able to explain the actions of the decision-maker.

For any node n_q, the conditional probability required to specify the Bayes's net is computed based on the relative importance (*weights*) of the parent nodes $pa(n_q)$ to itself.
If the state of n_q and pa(n_q) is the same, then $\Pr(n_q \mid \mathrm{pa}(n_q)) = \sum_{\mathrm{pa}(n_q)} (w_{pq} - (c-1)\kappa)$

$$\text{else } \Pr(n_q \mid \mathrm{pa}(n_q)) = \sum_{\mathrm{pa}(n_q)} \kappa \qquad (1)$$

where *c is the number of states and* $0 \le w_{pq} \le 1$.

κ is a constant and a measure of uncertainty such as careless errors, lucky guesses, changes in the student knowledge state due to learning and forgetting, and patterns of student responses unanticipated by the designer of the student model. The weights w_{pq} are either assessed based on the teacher's subjective judgment or past students' responses to closely related items.

Referring to Figure 3, since Learning_Objective_1 is dependent only on Evidence_1, w_1 = 1. Let Learning_Objective_1 has greater influence on mastery of goal "XYZ" than Learning_Objective_2, w_{1g} = 0.6, and w_{2g} = 0.4. Assigning κ= 0.005, the conditional probability tables can be computed using equation (1).

5 Tutoring Strategy

When a student logon to *iTutor*, the system automatically searches his ability index from the database. The ability index is either computed from the tests taken previously by the students, or from her knowledge states in the student model (see Section 5.1). Human tutors consider the student' s emotional state in deciding how to respond. Similarly in *iTutor*, the system considers factors such as response time, response pattern, student knowledge structure to determine tutoring actions: give more hint, help, ask another question, or stop the tutoring session. If the decision is to prompt another item, a learning objective and an appropriate item will be selected to coach her (see Section 5.2). Section 5.3 discussed the generation of tutoring strategy based on student' s response.

5.1 Mapping of Knowledge State to Student Ability

Let the student' s ability be $\theta_j = (\theta_{j1}, \theta_{j2}, ..\theta_{jm}, ..\theta_{jp})$. A function $f: v_{jm} \rightarrow \theta_m$ where v_{jm} is the evidence at the goal node (g) of m^{th} Bayes' s net. An example of such function is:

$$\theta_{jm} = \begin{cases} N(1.5, 0.6) & v_{jm} \geq 0.7 \\ N(0.5, 1) & 0.4 < v_{jm} < 0.7 \\ N(-1, 1.2) & v_{jm} \leq 0.4 \end{cases}$$

where $N(\mu, \sigma)$ denotes a *normal* distribution with mean μ and standard deviation σ

The computed ability index is then used to categorize (Advance, Intermediate, or Beginner) the student. An appropriate learning objective is selected based on the heuristic shown in Table 2. Value assignment is used to compute the path length of Bayes's net and is used as preference for tutoring policy generation. They are as follow:

$$\begin{aligned} &\text{value}(G) = 0 \text{ for } G \in \{\text{Goal nodes}\} \text{and ch}(G) = \phi \\ &\text{value}(\text{ch}(N)) = 0 \text{ if ch}(N) = \phi \\ &\text{value}(N) = \text{value }(\text{ch}(N)) + 1 \text{ for node } N \\ &\text{where ch}(N) \text{ is the child node of } N \end{aligned} \quad (2)$$

Table 2 Search Heuristic for Identifying Learning Objective for Coaching

Category	Identification of first Learning Objective for tutoring
Advance	25% of pathLength*
Intermediate	50% of pathLength
Beginner	75% of pathLength
New#	50% of pathLength ($\lceil$pathLength/2$\rceil$)

Note:
* pathLength denotes max(value(N)) $\forall N \in$ Bayes's net.
The value(N) assignment is stated in equation 2
The category *New* refers to students who login to the system the first time

5.2 Item Selection

Each item is tagged with an index (b_i) that estimates the minimum ability to answer it correctly with 0.5 probability. The items are assumed to be independent and the index obtained through statistic of past students' attempts or assigned using teacher' s belief. Subsequent update of item difficulty index may be performed through item response theory [4] such as Rasch model [10].

From the set of items related to a learning objective, an item i is selected based on: $\theta_j - b_i < \varepsilon$ where ε is a

Table 3 Utility of Various Outcomes

Condition / Expression	Preference
Decision: Stop	
$S(N)$ = "M" & value(N) = 0	1
$S(N)$ = "N" & value(N) = 0	0
k = number of $N_k \in \{(N_k, S_k)\}$ with same S_k	$1 - k/5$
number of tries, n, for the same learning objective	$1 - n/5$
Decision: Ask item on same N	
$S(N)$ = "M"	0
$S(N)$ = "N"	1
$S(N)$ = "P"	$\left(1 - \frac{\text{value(N)}}{\text{pathLength}}\right) \times \eta$ where η is a constant
Decision: Ask item on ch(N)	
$S(N)$ = "M"	1
$S(N)$ = "N"	0
$\gamma = \max(\Pr(S(\text{ch}(N)) = \text{"M"} \mid x = 1) - \Pr(S(\text{ch}(N)) = \text{"N"} \mid x = 0)$	
Decision: Ask item on pa(N)	
$S(N)$ = "M"	0
$S(N)$ = "N"	1
$\gamma = \max(\Pr(S(\text{pa}(N)) = \text{"M"} \mid x = 1) - \Pr(S(\text{pa}(N)) = \text{"}N\text{"} \mid x = 0))$	

Remarks: $S(N)$ denotes the knowledge state of node N
ch(N) denotes child node of node N

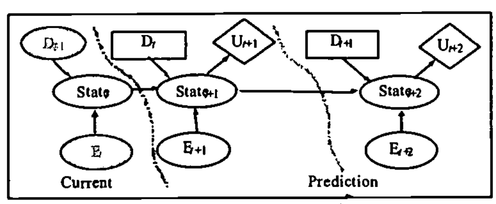


Figure 4: Dynamic Decision Network

Table 4 Utility Values for Item Difficulty Level Selection

Current Knowledge State ($State_t$)	Current Response (E_t)	Next Question Typ (D_{t+1}) Easy	Ave.	Diff
Non-Mastery	Correct	-0.2	1	0
	Wrong	0.4	-0.2	-0.4
Partial Mastery	Correct	-0.4	0.2	1
	Wrong	0.4	1	-0.2
Mastery	Correct	-0.4	-0.2	-0.2
	Wrong	-0.2	0.2	-0.2

pre-defined small value. This ensures selected item is challenging and likely to be solved by the student. Teacher's solution will be displayed upon student's request so that she can learn from her mistake. This strategy assumes student's ability is dynamic and can be raised to higher levels through self-paced computer-aided tutoring.

5.3 Tutoring Policy Generation

To bring the probabilistic network one step closer to being a useful *intelligent tutoring system*, automated decision-making capability has been added. When asked to provide a tutoring policy for the student, the system generates a course of action based on her current mastery states. The tutoring policy aims to use a series of items with differing difficulty to determine more precisely her mastery of specific learning objectives. Items are categorized into *easy*, *average* and *difficult*. In this project, a two-step look-ahead dynamic decision network is recommended so as to compromise between the need to invoke policy generation routine for a decision and the long computing time to generate policy with many decisions.

Define a Bayes's net with nodes N,
E_t be the evidence at time t,
Bel(N) be a vector of probabilities (updated over time),
d_t be the decision at time t, and
α be the normalization constant.

Prediction 1: $\mathrm{Bel}^*(N_{t+1}) \leftarrow \sum_{N_t} \Pr(N_{t+1} \mid N_t, d_t)\mathrm{Bel}(N_t)$

Estimation 1: $\mathrm{Bel}(N_{t+1}) \leftarrow \alpha \Pr(E_{t+1} \mid N_{t+1})\mathrm{Bel}^*(N_{t+1})$

Prediction 2:
$\mathrm{Bel}^*(N_{t+2}) \leftarrow \sum_{N_{t+1}} \Pr(N_{t+2} \mid N_{t+1}, d_{t+1})\mathrm{Bel}(N_{t+1})$

Estimation 2: $\mathrm{Bel}(N_{t+2}) \leftarrow \alpha \Pr(E_{t+2} \mid N_{t+2})\mathrm{Bel}^*(N_{t+2})$

Expected Utility : $\sum_{N_t} \mathrm{Bel}(N_t) \max_{d_t} \sum_{N_{t+1}} \Pr(N_{t+1} \mid N_t, d_t) \times$
$\max_{d_{t+1}} \sum_{N_{t+2}} \Pr(N_{t+2} \mid N_t, d_t, N_{t+1}, d_{t+1}) \times$
$U(N_t, d_t, N_{t+1}, d_{t+1}, N_{t+2})$

Figure 5: Prediction-estimation Process

Figure 4 shows a dynamic decision network (DDN) used in this study. In addition to the decision nodes for current and future time steps, the DDN also contains the previous decision, D_{t-1}, as an evidence node. When the evidence for state t arrives, the probability distributions of $State_t$ are updated [1] using the prediction-estimation process (see Figure 5). After the initial prediction of probabilities (Bel*), $State_{t+1}$ estimates the new belief based on projected evidence [13]. This process repeats for $State_{t+2}$. Eventually, the expected utility is evaluated by a sequence of summations and maximizations. Tables 3 and 4 show the utility functions for node U_{t+2}. Selecting the outcomes with maximum expected utility value constitute the tutoring policy.

6 An Illustration

6.1 Construction of a Decision Network

In this project, the construction of all probabilistic networks is performed using Netica API [7]. A module leader enters the learning objectives and the weights of the key concept *Forces* using Microsoft Access [6]. The probabilistic values shown in Figure 6 are entered based on past examination results. By clicking the button "Model Construction", a Bayes's net (see Figure 7) and a decision network (see Figure 8) on "Forces"

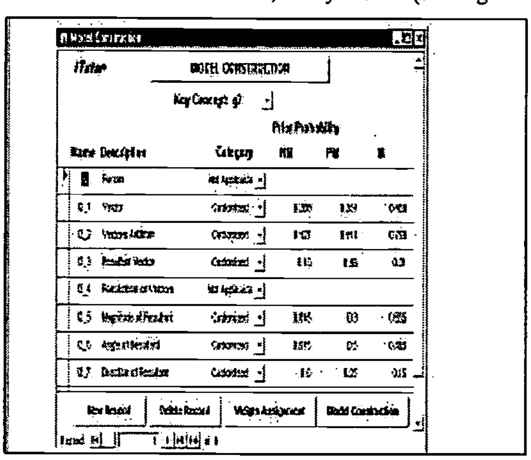


Figure 6: A Snapshot on Data Entry for Model Construction

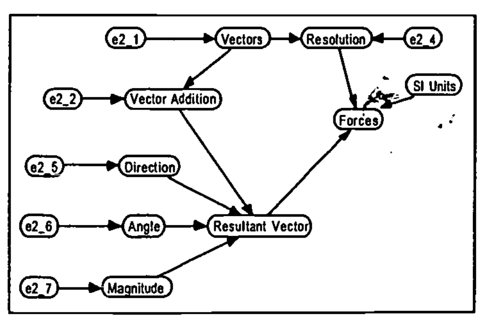


Figure 7: Bayes's Net on *Forces*

will be created. Teachers who are familiar with Netica application [8] can use the generated Bayes's net to perform *what-if* analysis. For example, a teacher may want to determine the likely student's improvement if he provides remedial instructions on "Resolutions of Vectors". He can do so by instantiating the evidence node *e2_4* to "Mastery" state, and observe the probability of mastery in the goal node labeled *Forces*.

6.2 Diagnosis of a Student's Misconceptions

The items to be presented to the students are coded by the teacher using Scientific Notebook [15]. With *iTutor*, the teacher is able to monitor student's progress through the database management tool. Figure 9a shows a snapshot of a student who had answered item "Force_001" correctly and partially correct for item "Force_004". The teacher can track a student's mastery states by clicking the "Advice" button. The system transforms the responses to evidence, and instantiates the evidence nodes in the Bayes's net as shown in Figure 9b. The posterior mastery states are displayed (see Figure 9c). The output also provides the teacher information on specific learning objectives to tutor. In addition, he can also examine the detailed strategy by clicking the "Tutorial Strategy" button. This action causes the generation of a decision network (see Figure 9d). Figure 9e shows items to be posed to the student if she continues with the online tutorial. At any stage, the teacher may intervene by providing personal coaching.

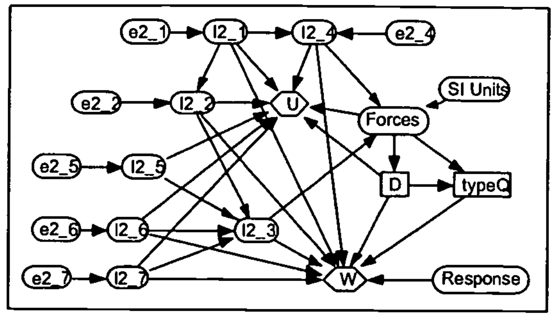


Figure 8: Decision Network on *Forces*

7 Conclusions

Presently, the students' knowledge states remain unchanged until additional evidence is available. The system also uses a constant learning rate for all students. One future direction is to include additional parameters to model student forgetting and learning rates. Another area is to provide a user interface for teachers not familiar with Netica application to perform *what-if* analysis. In this way, the teacher will be able to focus on student's issues rather than to learn another software tool. The next future direction is to include probability functions other than *Normal* distribution. This is essential when the ability distribution of student cohort is not symmetric.

A significant result of this project is the use of Bayesian networks to generate sound probabilistic inferences. Another contribution is the automation of decision networks construction. The recommended strategy is used in adaptive tutoring. With *iTutor*, teacher is able to monitor the student's progress and yet had time for lesson preparation and coaching of weaker students. In addition, the teacher has accessed to the student's knowledge states and actions taken by *iTutor* at every stage of the tutoring process. Moreover, it enables students to have tutorials customized to their needs.

References

[1] Castillo, E., Gutiérrez, J. M., & Hadi, A.S., "Expert Systems and probabilistic network Models", Springer-Verlag, (1997).

[2] Conati C. & Vanlehn K., "Teaching meta-cognitive skills: Implementation and evaluation of a tutoring system to guide self-explanation while learning from examples", Proceedings of AIED '99, 9th World Conference of Artificial Intelligence and Education, Le Man, France, (1999).

[3] Deam, T. L. & Wellman, M. P., "Planning and Control", Morgan Kaufmann, San Mateo, California, (1991).

[4] Hambleton, R.K., Swaminathan, H., & Rogers, H. J., "Fundamentals of Item Response Theory", Sage Publications, Inc, (1991).

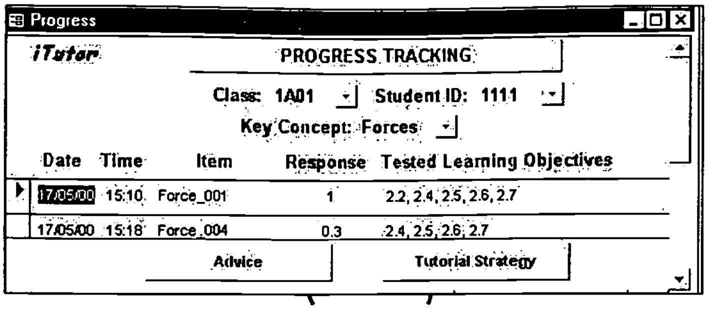
[5] Howard, R.A. & Matheson, J.E. "Readings on the Principles and Applications of Decision Analysis", Menlo Park, California: Strategic Decisions Group, (1983).

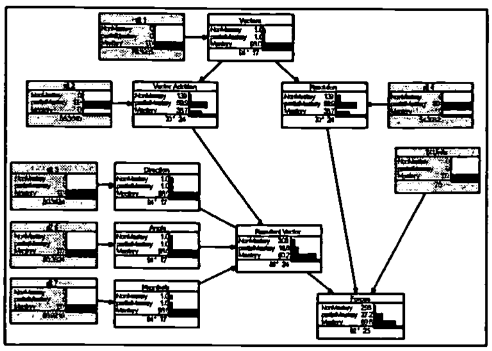
[6] Litwin, P., Getz, K., & Gilbert, M. , "Access 97 Developer's Handbook", 3rd edition, Sybex Inc, (1997).

[7] Norsys, "Netica API Programmer's Library Reference Manual", version 1.06, Norsys Software Corp., (1999)

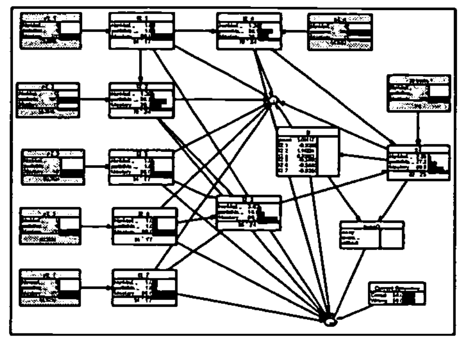
[8] Norsys, "Netica Application Reference Manual", version 1.12, Norsys Software Corp., (1999).
[9] Pearl, J., "Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference", Morgan Kaufmann Publishers, Inc, (1988).
[10] Rasch, G., "Probabilistic Models for Some Intelligence and Attainment Tests", Danish Institute for Educational Research, (1960).
[11] Reye, J., "A belief Net Backbone for Student Modeling, Proceedings of Intelligent tutoring Systems", Third International Conference, ITS '96, Springer-Verlag, (1996).
[12] Ross, S., "A First Course in Probability", Prentice-Hall International, Inc., 5th edition, (1998).
[13] Russell, S. & Norvig, P. , "Artificial Intelligence: A Modern Approach", Prentice Hall, (1995).
[14] Simon H.A., "The New Science of Management Decisions", New York: Harper & Row, (1960).
[15] TCI Software Research, "Scientific Notebook version 3.00 Build 939", Brooks / Cole Publishing Company, (1998).
[16] Villano, M., "Probabilistic Student Models: Bayesian Belief Networks and Knowledge Space Theory", Proceedings of Intelligent Tutoring Systems: Second International Conference, ITS '92, Springer-Verlag, (1992)

(a) User interface for teacher to track student's progress





(b) Bayes's net running as background process (transparent to user)



(d) Dynamic decision network running as background process (transparent to user)

Student ID: 1111

The student's mastery states are :

Learning Objective		NonMastery	Partial	Mastery	Value
12_1	Vectors	0.010	0.010	0.980	93.75
12_2	Vector Addition	0.014	0.599	0.387	69.87
12_5	Direction	0.010	0.010	0.980	93.75
12_6	Angle	0.010	0.010	0.980	93.75
12_7	Magnitude	0.010	0.010	0.980	93.75
12_3	Resultant Vector	0.030	0.168	0.802	85.71
12_4	Resolution	0.014	0.599	0.387	69.87
g1	SI Units	0.000	0.000	1.000	95.00
g2	Forces	0.030	0.272	0.698	81.59

The expected score for this key concept Forces is 81.59.

Based on the knowledge states, you may want to provide coaching in Vector Addition, and Resolution.

(c) Output of student's mastery states

Student ID: 1111
With regard to the key concept Forces, the course of action is :

```
select average item from 12_2 (Force_002)
 if response is correct then
    select difficult item from 12_2 (Force_012)
    if response is correct then
        select average item from 12_4 (Force_013)
    else
        select average item from 12_2 (Force_021)
 else
    select easy item from 12_2 (Force_003)
    if response is correct then
        select average item from 12_2 (Force_017)
    else
        select easy item from 12_2 (Force_006)
```

(e) Output of tutoring strategy

Figure 9: Overview of an *iTutor* Session

NOTICE

REPRODUCTION BASIS

☒ This document is covered by a signed "Reproduction Release (Blanket) form (on file within the ERIC system), encompassing all or classes of documents from its source organization and, therefore, does not require a "Specific Document" Release form.

☐ This document is Federally-funded, or carries its own permission to reproduce, or is otherwise in the public domain and, therefore, may be reproduced by ERIC without a signed Reproduction Release form (either "Specific Document" or "Blanket").