Int. J. Engng Ed. Vol. 10, No. 6, pp. 496–501, 1994
Printed in Great Britain.

0742–0269/91 $3.00+0.00

Evaluation and Classification of Learner Knowledge Levels for a CAL Course*

MICHAEL S. WALD
BOGDAN SOLTYSIAK
Hamburg Educational Partnership, Berliner Tor 21, 20099 Hamburg, Germany

With the dynamic development of computer technology, the role of the computer in the learning process is rapidly growing. These developments come from both hardware (new mutimedia technology) and software (authoring software). New technologies for presenting learning materials require new learning strategies. A design for an intelligent questioning module that can be used before the learner starts a real CAL (computer-aided learning) programme is presented. This module can evaluate the learner's prior knowledge and locates his/her starting knowledge level for different information frames in a single lesson. We are testing learner knowledge about facts (information units), as well as the ability to conclude and to solve problems. Afterwards the learning programme automatically adapts itself to the learner's level and then follows an individual path through the subject matter. Based on this, a prototype learning system containing introductory materials for an electronics module has been designed.

INTRODUCTION

RESEARCH in intelligent tutoring systems (ITS) is based on two main strategies: the learner model and the teacher model. The learner model in ITS provides support for the following functions: charting the learner's progress through the curriculum, selecting the appropriate level of hinting and explanation, and facilitating learner feedback [1].

We present one possible way of constructing a learner model before embarking on a computer-aided learning (CAL) course. It is a part of our ITS system described in a previous paper [2]. This model assists the learner in selecting an optimal starting point and an individualized learning path in a CAL course within a well-defined curricular framework. The system tests elementary knowledge with questions, and problem-solving ability with tasks to solve.

Knowledge in the context of this paper encompasses the capacity of absorbing and processing information for the application and the development of new concepts. The system also monitors learner's questions to the system.

A test structure is presented that can be adopted for all subject matter. The software with minor modifications and with a different knowledge base can be used for generic learner evaluation purposes.

* Accepted 30 August 1994.

STRUCTURE OF THE CAL COURSE

The material of the course is divided into lessons. The number of lessons is determined by the course designer and by the demands of the subject matter.

During the course, lessons appear sequentially, as is typical for a linear learning process. Every lesson consists of information frames called frames, as shown in Fig. 1. The frames are represented at different levels of expertise. In [2] we selected four levels for consideration—basic, middle, advanced and expert. These levels are meant to correspond, for example, to commonly implemented learning levels such as school, technician, engineer and researcher.

In the previous paper [2] we presented a course unit on junction transistors as a part of an 'Introduction to Electronics' programme, which was divided into three lesson units.

In order to complete the whole course at a given level, the learner should follow a predefined sequence of lessons. This means that he/she will not start on transistor circuit applications before mastering transistor properties.

The initial basic lesson in the course is 'Transistor Characteristics'. This lesson with which we will deal here consists of three frames:

- Transistor construction and operation.
- Common-base configuration.
- Common-emitter configuration.

In order to proceed with the course, the learner has to master the contents of this lesson to an acceptable level as determined by the subject expert. Prior to starting the first lesson, the programme evaluates on request relevant prior knowledge, detecting the areas in which the learner is too

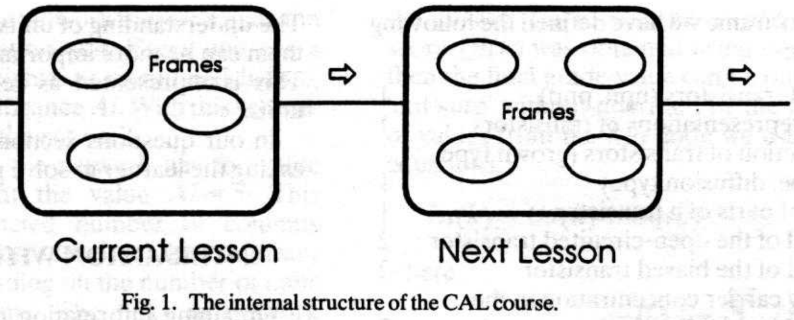

Fig. 1. The internal structure of the CAL course.

weak to continue meaningful progress without resorting to additional work.

The program then assembles available resources and materials, such as video and other instructional resources, for each frame at the appropriate learner level, creating a unique path through the subject matter responding to the learner's assessed needs.

In creating the path we note that the learner's prior knowledge level related to the different frames need not be compatible. As is often the case, areas of deeper or better digested knowledge may be interspersed with weaker areas.

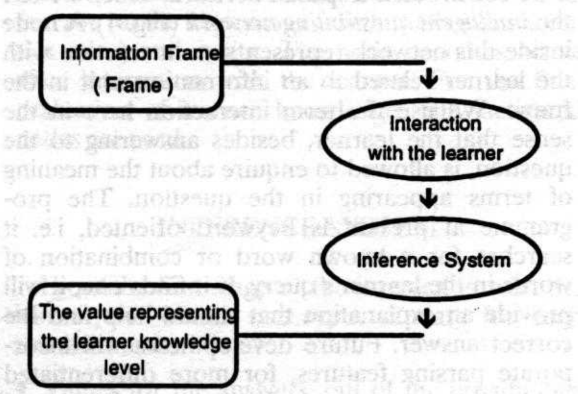

Fig. 2. Functional elements in the assessment process of learner knowledge within one frame.

In order to obtain an assessment of learner knowledge, at the beginning of each lesson, we invite him/her to pass a test. At the end of this test, we obtain a numerical value corresponding to the learner's knowledge level for each frame in the test (Fig. 2).

The four levels—basic, middle, advanced and expert—are represented by numerical values. These values are used in the tutorial to construct an individualized learning path for the learner.

THE INFORMATION FRAME

We note that every frame represents facts (information units) and relations between them (Fig. 3). A frame can be represented graphically as consisting of nodes and links. This representation is commonly used by designers of expert systems. Nodes are generally used to represent physical objects, concepts or situations. The links are used to express relationships [3]. Following this idea we can say that every lesson consisting of frames can be also represented in this manner.

Now, we will concentrate on a single frame. A domain expert should prepare for all frame materials to be presented in the tutorial part, as well as questions about units and relations between them for the questions part.

In our example, in the 'transistor construction

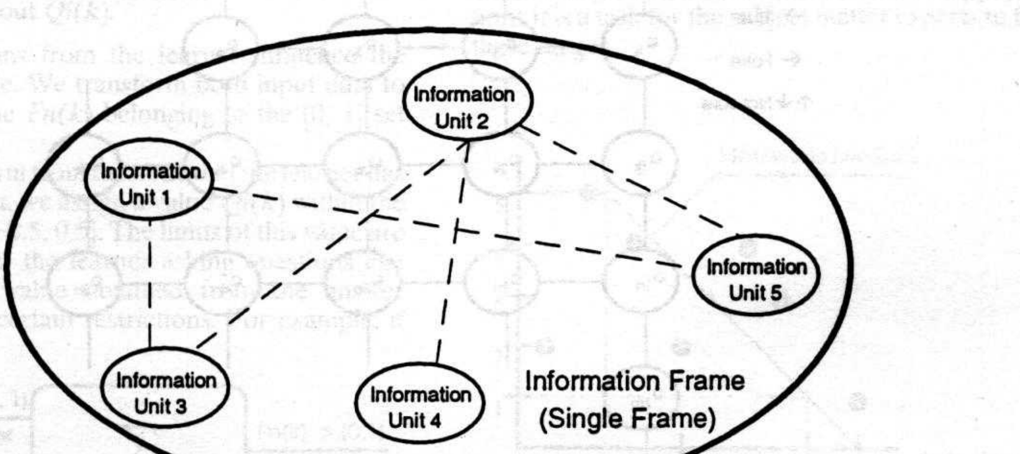

Fig. 3. Graphical representation of a single frame.

and operation' frame we have defined the following types of units:

1.	Types of transistors (npn, pnp)	1
2.	Circuit representations of transistors	1
3.	Construction of transistors (grown type, alloy type, diffusion type)	1
4.	Names of parts of a transistor	1
5.	Potential of the open-circuited transistor	2
6.	Potential of the biased transistor	2
7.	Minority carrier concentration in the open-circuited transistor	3
8.	Minority and majority carriers in the biased transistor	3
9.	Transistor biased in the active region	3
10.	Elements of the emitter current in the biased transistor	4
11.	Elements of the collector current in the biased transistor assuming $I_e = 0$	4
12.	Collector current in the biased transistor	4
13.	Transistor as a voltage amplifier	5
14.	Transistor as a current amplifier	5

The scores on the right represent the significance and complexity of a certain unit according to the domain expert. The numbers also relate to predefined information areas, as set by the subject expert. We allow the expert to set five levels of significance.

All units are important for comprehensive information about the frame, but some of them can be considered as more important for the learner's progress than others within the whole lesson. For example, mastery of unit (4), probably implies that the learner also knows number (2) but in reverse order it is unlikely.

One can proceed with the lesson unhindered without knowing about some units with a score of 2, when one knows about units with a score of 4, but not vice versa.

In addition to information on the contents of units, the learner needs to apply the information. The understanding of units and the ability to apply them can be more important than just information. This is represented as relations (links) between units.

In our questions section we are testing that by asking the learner to solve problems and tasks.

INTERACTION WITH THE LEARNER

Obtaining information on the learner's knowledge level is accomplished in two stages.

In the first stage, in order to get information about the learner's comprehension of knowledge contained in the units, we ask the learner questions and analyse possible questions from the learner.

The second stage is based on results from the first; we test, through tasks, understanding and the ability to make conclusions.

The second stage is optional and depends and is controlled by the teacher or domain expert. To benefit from this second part the expert should set problems, as well as the rules, to determine the questioning sequence generated by the answers.

Let us start with testing the learner's knowledge of the basic electronics units presented above. We have constructed a special network, which we call the *intelligent questioning network* (Fig. 4). A node inside this network represents an interaction with the learner related to an information unit in the frame. We use the term interaction here in the sense that the learner, besides answering to the question, is allowed to enquire about the meaning of terms appearing in the question. The programme at present is keyword oriented, i.e. it searches for a known word or combination of words in the learner's query. If it finds one, it will provide an explanation that should help find the correct answer. Future developments will incorporate parsing features, for more differentiated approaches.

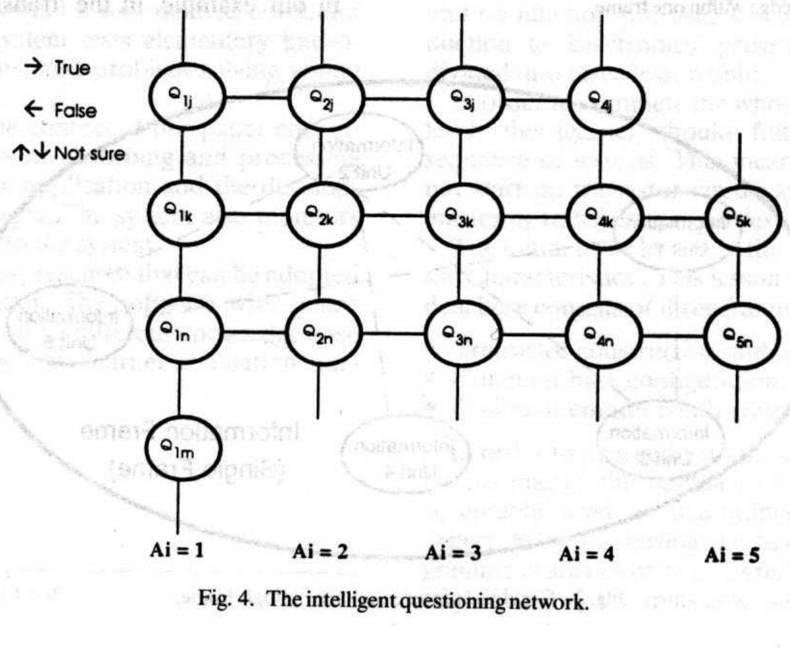

Fig. 4. The intelligent questioning network.

For every node in the network there is a question related to the information unit. Every unit has a significance value (weight), as mentioned before, that represents its significance Ai. With this scheme we insert questions in the network.

The left column of the network has the value $Ai = 1$, and the right the value $Ai = 5$. This network has a restricted number of columns related to the significance levels, and an arbitrary number of rows depending on the number of units existing within a frame. The links in network determine the next question to ask, related to the learner's response aggregate of answers and questions from the previous interaction node. The whole interaction starts from a question chosen at random at a middle level, with the significance value $Ai = 3$.

The evaluation coming from the interaction with the learner from one node can have three values: 'true', 'false', and 'not sure'. A 'not sure' result causes vertical movement, along the column, where the questions have the same level of significance. 'True' and 'false' cause a horizontal displacement along the row, to the right and left, respectively. Using this scheme we are able to assess the range of units a learner knows well.

Every node in the network has two states: *inactive* (at the beginning of the test all nodes are in this state) and *active*. The node becomes active after interaction with the learner. For the final evaluation of learner knowledge, only activated nodes are used.

INFERENCE SYSTEM

From the first stage of the interaction with the learner, we obtain two types of data for each activated node:

- Values for the answers, out of the possibilities $Ua(k) = \{0, 0.5, 1\}$, representing 'false', 'not sure', and 'true' answers respectively.
- Information about which terms the learner has enquired about $Qi(k)$.

The questions from the learner influence the score at a node. We transform both input data to the single value $Fn(k)$ belonging to the $\{0, 1\}$ set (Fig. 5).

For every term from the node that the learner has enquired about, we assign a value $Qi(k)$ within the set of values $\{-0.5, 0.5\}$. The limits of this value are so defined that the learner asking questions can influence the value obtained from the answer $Ua(k)$ within certain restrictions. For example, if the answer $Ua(k)$ was 'false', and the maximum score $Qi(k)$ was obtained using the query option, then the final grade value can be only as good as a 'not sure' without queries. For the transformation of values from the kth node we use the following equations.

$$Fn(k) = Ua(k) + Uq(k)$$

where

$$Uq(k) = Qi(k) \begin{cases} -0.5 \text{ for } \sum_{i=0}^{i=n} Qi(k) < -0.5 \\ 0.5 \text{ for } \sum_{i=0}^{i=n} Qi(k) > 0.5 \end{cases}$$

but

$$0 \leqslant Fn(k) < 1$$

After obtaining the values Fn for all activated nodes, we apply concepts from fuzzy logic theory [4, 5] and represent them in two-dimensional space (Fig. 6). On the horizontal axis we plot $Fn(k)$; on the vertical axis the significance value Ai. Analogous to the fuzzy set theory we define the membership function μ. The meaning of this function, represented here as a line is as follows. When the point that represents the value from an interaction node lies above this line, we accept this value as indicating that the learner is sufficiently proficient with the relevant information unit to proceed to the next line of difficulty.

Figure 6 shows that for evaluation from the interaction mode with $Ai = 1$ the learner should get at least a value of 0.86, corresponding to the $Ai = 5$ at least value 0.3. According to this scheme, we have set that if >50% of the activated notes lie below the membership line, then the learner's knowledge within the frame is on the basic level only. For the middle level, in our example, >50% should lie above the line, 70% above for the advanced level, and 90% above for the expert level. This distribution is arbitrary, and is only described as an example of the method. In real-world applications it is a task for the subject matter experts to fix these ranges.

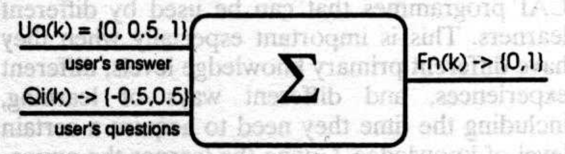

Fig. 5. Transformation of data from the kth question node.

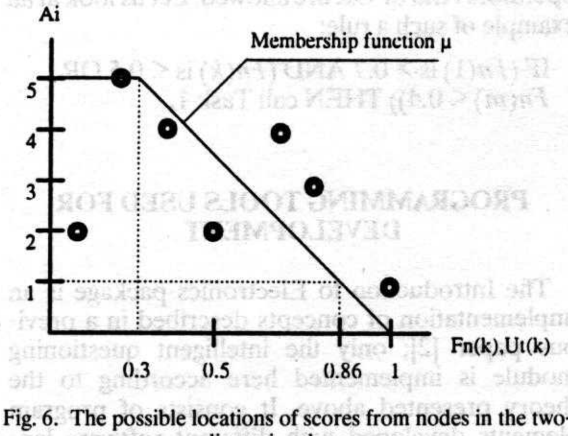

Fig. 6. The possible locations of scores from nodes in the two-dimensional space.

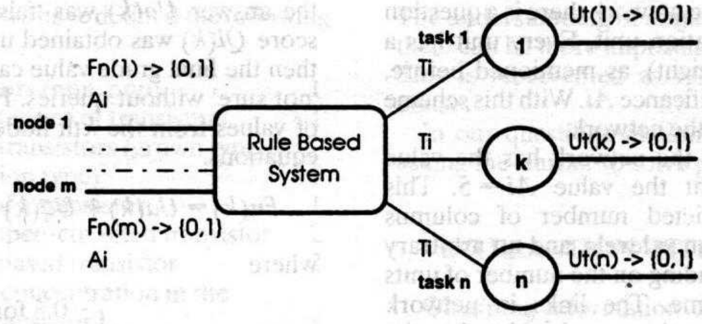

Fig. 7. Test of learner reasoning and comprehension ability.

Let us now imagine that we would like to test the learner's ability to apply information units, i.e. comprehension and reasoning. This stage is optional; it can be included or excluded by the subject matter expert in our scheme.

Within this second comprehension and reasoning stage, having placed the learner within a certain range of information units (activated nodes), based on a rule system, he/she will be given some tasks (Fig. 7).

Every task has its own significance value *Ti* parallel to the significance value *Ai* for the information unit. As a result, from each solved task we get a value $Ut(k)$ belonging to the $\{0, 1\}$ set. We are using these results as additional information for the final learner evaluation pertaining to a single frame. We add these second stage result values to the two-dimensional space diagram that was presented before (Fig. 6). According to that, to the horizontal axis we have to add the second parameter $Ut(k)$. Other definitions remain the same.

The tasks have also been divided into five significance levels (groups), parallel to the significance levels of information units. Every task is associated with a unique logical rule. The task is presented to the learner when the rule is true. The program checks all rules, one after another, starting from the tasks that have the lowest significance value.

The rule has the form of IF (*expression*) THEN (*call a task*). Within the expression of the logical operators And or OR are allowed. Let us look at an example of such a rule:

IF ($Fn(1)$ is > 0.7 AND ($Fn(k)$ is < 0.5 OR $Fn(m) < 0.4$)) THEN call Task 1.

PROGRAMMING TOOLS USED FOR DEVELOPMENT

The Introduction to Electronics package is an implementation of concepts described in a previous paper [2]; only the intelligent questioning module is implemented here according to the theory presented above. It consists of program elements developed with different software languages and tools. These elements are specific for each goal, such as simulation or tutorial, and it would be difficult and time consuming, and in some cases impossible, to apply a single tool to design a whole package.

For example, one could design a whole system using C or C++, but the development time needed for that would be too long. It would be also very difficult to design an interesting tutorial with multimedia futures using a higher-level, object-oriented programming language like KnowledgePro for Windows [6]. Finally, it would not be possible to design an expert system or a simulation using an authoring tool such as Authorware Professional for Windows [7].

After many tests concerning interfaces and the data exchange between different software modules, we have decided, for programming different parts of the package, to use all the tools mentioned above.

The intelligent questioning module and the controlling module are written in KnowledgePro. The simulation of the transistor circuit and the module that controls the learning loop with C++. The tutorial is designed with Authorware Professional. Introduction to Electronics works under Windows 3.1 with 256-colour graphics and on at least a 80486 PC machine.

CONCLUSIONS

Current developments in new learning media offer novel ways of analysing and designing new intelligent evaluation procedures, and new ways of employing the usefulness of CAL/CAI programs in the learning process. Classrooms are overcrowded, instructors are stretched to cater for personal needs of the learner. With this paper we would like to point out the need to design CAL/CAI programmes that can be used by different learners. This is important especially when they have different primary knowledge levels, different experiences, and different ways of learning, including the time they need to acquire a certain level of knowledge. Giving the learner the opportunity to test prior knowledge and then follow an individualized path seems to us a promising idea for future developments in this area. We are thinking especially of those who would like to learn at home, or in courses where learners have different backgrounds, e.g. language courses. The proposed scheme avoids accidental evaluations and points to the possibility of evaluation which is not easy to accomplish in common tests.

Acknowledgements—The authors would like to record their gratitude to the European Union's COMETT Program for partial support of this work.

REFERENCES

1. M. Villano, Probabilistic student models: Bayesian belief networks and knowledge space theory. *Proc. 2nd International Conference on ITS 92*, Montreal, pp. 491–498 (1992).
2. M. S. Wald and B. Soltysiak, Frame oriented intelligent tutoring system, *Proc. International Conference on Computer Aided Engineering Education*, Bucharest, pp. 419–429 (1993). Also in *Int. J. Engng Ed.*, **10**, 215–221 (1994).
3. J. Girratano, *Expert Systems: Principles and Programming*. PWS-KENT, Boston, MA (1989).
4. D. Brubaker, *Fuzzy-Logic Basic: Intuitive Rules Replace Complex Math*. Cahners, The Netherlands (1992).
5. T. Terano, K. Asai and M. Sugeno, *Fuzzy Systems Theory and its Applications*. Academic Press, San Diego, CA (1992).
6. KnowledgePro for Windows, Knowledge Garden Inc. (1991).
7. Authorware Professional for Windows, Authorware Inc. (1991).

Bogdan Soltysiak has a degree in information science from the Technical University of Wocław. He has been with the Hamburg Educational Partnership with a DAAD and COMETT grant. He is currently taking up a teaching position at the Technical University of Wrocław.

Michael S. Wald is editor of the journal. He is also executive director of the University Enterprise Institute of the Fachhochschule of Hamburg and of the Hamburg Educational Partnership.