PERGAMON

Computers & Education 40 (2003) 217–235

COMPUTERS & EDUCATION

www.elsevier.com/locate/compedu

A conceptual map model for developing intelligent tutoring systems☆

Gwo-Jen Hwang*

Information Management Department, National Chi Nan University, Pu-Li, Nan-Tou, Taiwan 545, ROC

Received 15 October 2001; accepted 14 August 2002

Abstract

With the recent rapid progress of computer technology, researchers have attempted to adopt artificial intelligence and use computer networks to develop computer-aided instruction systems. Meanwhile, researchers have also attempted to develop more effective programs to test and enhance the learning performance of students. However, conventional testing systems simply give students a score, and do not give them the opportunity to learn how to improve their learning performance. Students would benefit more if the test results could be analyzed and hence advice could be provided accordingly. This study proposes a conceptual map model, which provides learning suggestions by analyzing the subject materials and test results. A testing and diagnostic system is also implemented on computer networks based on the novel approach. Experimental results have demonstrated that the novel approach benefits students and deserves further investigation.

Keywords: Computer-assisted instruction; Learning diagnosis; Test bank; Conceptual map

1. Introduction

With accelerated growth of computer and communication technologies, researchers have attempted to adopt computer network technology for research on education. Notable examples include the development of computer-aided tutoring and testing systems (Hopper, 1992). In 1989,

☆ This study is supported by the National Science Council of the Republic of China under contract numbers NSC-90-2520-S-260-001 and NSC-90-2520-S-260-002.

* Tel.: +886-915396558; fax: +886-49-2915205.
E-mail address: gjhwang@ncnu.edu.tw (G.-J. Hwang).

PII: S0360-1315(02)00121-5

Johnson et al. proposed a software design and development research program called Microcomputer Intelligence for Technical Training (MITT). Specifically, Johnson et al. presented MITT Writer, an authoring environment for building intelligent tutoring systems for computer courses, which represented a practical application of artificial intelligence (AI) in technical training (Johnson, Neste, & Duncan, 1989). In the same year, Vasandani et al. developed an intelligent tutoring system that helps to organize system knowledge and operational information to enhance operator performance (Vasandani & Govindaraj, 1991, 1995; Vasandani, Govindaraj, & Mitchell, 1989). Meanwhile, Gonzalez and Ingraham designed an intelligent tutoring system, which is capable of automatically determining exercise progression and remediation during a training session according to past student performance (Gonzalez & Ingraham, 1994). Harp et al. employed the neural networks technique to model student behavior in the context of an intelligent tutoring system. This approach used self-organizing feature maps to capture possible student knowledge states from an existing test database (Harp, Samad, & Villano, 1995). Furthermore, Rowe and Galvin employed planning methods, consistency enforcement, objects and structured menu tools to construct intelligent simulation-based tutors for procedural skills (Rowe & Galvin, 1998). Hwang also proposed an intelligent tutoring environment which can detect the on-line behaviors of students (Hwang, 1998). Meanwhile, Giraffa et al. demonstrated how a multi-agent systems (MAS) approach can be used to build an interactive intelligent tutoring system (Giraffa, Mora, & Viccari, 1999). In 2000, Ozdemir and Alpaslan presented an intelligent agent for guiding students through on-line course material. This agent can help students to study course concepts by providing navigational support according to their knowledge level (Ozdemir & Alpaslan, 2000). Clearly, the development of intelligent tutoring systems and learning environments has become an important issue in both computer science and education (Pugliesi & Rezende, 1999; Rosic, Slavomir, Stankov, & Glavinic, 2000; Wong, Quek, & Looi, 1998; Yoshikawa, Shintani, & Ohba, 2000).

Besides the growth of intelligent tutoring systems, the development of on-line testing systems has also attracted the attention of researchers. Taking GRE (Graduate Record Examinations) as an example, students have taken this test on computers since 1992, and the paper-and-pencil form was totally abandoned in 1999. IBM and Arthur Andersen have been working on developing computerized testing systems, and systems which replace traditional paper-and-pencil testing systems with on-line testing are proliferating rapidly.

In New Zealand, three researchers developed a ‘‘Knowledge Based Computer Assisted Instruction System’’ which can change the numerical part of test items while the test is in progress to prevent students from memorizing the answers (Fan, Tina, & Shue, 1996). Another branch of relevant research is Computerized Adaptive Testing, which applies prediction methodologies to reduce the length of the test without sacrificing accuracy (Wainer, 1990).

Conventional testing systems represent the learning status of a student by assigning that student with a score or grade. This approach makes the student aware of his/her learning status through the score or grade, but the student might be unable to improve his/her learning status without further guidance. The teacher can give students additional suggestions to improve their learning performance after the test. However, it is time-consuming for a teacher to give personalized suggestions to each student, particularly when the number of the students in the class exceeds 20. Therefore, intelligent testing system could be very helpful to teachers and students for identifying learning problems.

This study proposes a novel approach to cope with these problems. The relationships between subject concepts and test items are determined by analyzing the subject materials and the item bank, and the learning problems of each student are then identified based on these relationships. A testing and diagnostic system based on this approach is implemented, in which different test items are given even if an identical subject unit is tested repeatedly. This system can provide objective assessments and personalized suggestions for each student by analyzing student answers and the relationships among the subject concepts and the test items.

2. Relevant research

The rapid development of computer networks is allowing access to information and communication free of spatial and temporal constraints. Network communications allow problems to be solved through on-line discussion, and thus the implementation of computerized testing and practice systems has become an important issue.

In 1997, a testing and diagnostic system, ITES, was proposed (Hsu, Tu, Yeh, Chu, & Hwang, 1997; Hwang, 1999). ITES originated from the CORAL (Cooperative Remotely Accessible Learning) project in Taiwan (Sun & Chou, 1996). The project was initiated by a research group at National Chaio Tung University and involved eight sub-tasks:

1. The investigation of network-based tutoring systems, including pattern recording, remote data retrieval and access control.
2. The investigation of network learning environments, including real-time monitoring and tutoring process control.
3. The investigation of wide area network CALs, including feasibility, scalability and architecture.
4. The testing and evaluation of network-based CALs, including motivation and cognition analysis.
5. The investigation of interface design for network-based CALs, including screen layout, icon/window design and knowledge visualization.
6. The investigation of student modeling for network-based CALs, including the analysis of hypertext navigation and communication patterns.
7. The investigation of knowledge-based systems for tutoring process control, including knowledge representation for student characteristics and communication parameters, real-time analysis of student behavior and dynamic arrangement of tutoring schedule.
8. The investigation of interaction pattern analysis, including the analysis of social context.

The ultimate goal of the project is to accomplish a CAL system to help learning progress by developing an intelligent tutoring environment on computer networks. The following sections describe the learning diagnosis strategy of ITES, and also provide some experimental results to evaluate the efficacy of the novel approach.

3. Strategies of diagnosing learning status

During tutoring, students learn new concepts and new relationships among previously learned concepts, and this knowledge can be represented as a conceptual map (McAleese, 1994, 1998).

Salisbury indicated that learning information, including facts, names, labels, or paired associations, is often a perquisite to efficiently performing a more complex, higher level skill (Salisbury, 1998). For example, the names and abbreviations of chemical elements and their atomic weights must be thoroughly learned to comprehend scientific writings or chemical formulae. That is, relationships exist that indicate the effect of learning one concept on the learning of other concepts. Such relationships are called ''concept effect relationships'', and the following discussion presents such conceptual maps diagrammatically as ''concept effect graphs''.

3.1. Structure of subject materials and the conceptual map model

Subject materials can be viewed as a tree diagram comprising chapters, sections, sub-sections and key concepts to be learned (see Fig. 1). This approach offers an overall cognition of the subject contents, but additional information is required to diagnose student learning status. For example, if a student fails to learn the concept ''common divisor'', this may be because he/she did not learn the concept ''factors'' well. In this case, we would suggest that the student study ''factors'' more thoroughly before attempting ''common divisor''. That is, when the relationships among those concepts are identified, it is possible to determine the learning problems of individual students and provide suggestions.

To model these learning effect relationships among concepts, a conceptual map-based notation is proposed, namely concept effect *Relationships*. Consider two concepts, C_i and C_j, if C_i is perquisite to efficiently performing the more complex and higher level concept C_j, then a concept effect relationship $C_i \rightarrow C_j$ exists. A single concept may have multiple perquisite concepts, and can also be a perquisite concept of multiple concepts.

For example, the concept ''addition'' must be learned before ''multiplication''. Likewise, ''multiplication'' and ''subtraction'' must be learned before learning the concept ''division''. Fig. 2 presents the concept effect relationships for the subject unit ''numbers''.

3.2. Constructing the concept effect graph

To construct the concept effect graph, the relationships among the concepts to be learned are represented by a two-dimensional table, namely, the *concept effect table* (CET). Consider the

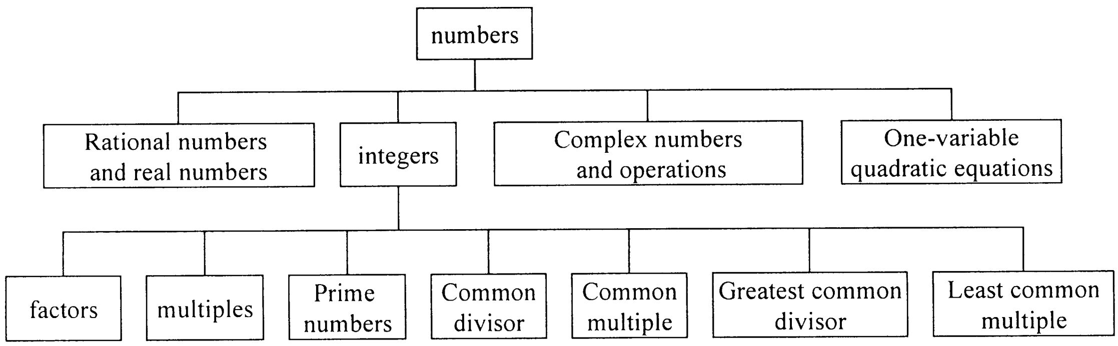

Fig. 1. Tree structure for ''numbers''.

example presented in Table 1. C_i represents the possible prerequisite concept of C_j, while NP_j represents the number of prerequisite concepts of C_j. If $CET(C_i,C_j)=1$, it is said that "C_i is one of the prerequisites of C_j". One possible reason for a student failing to learn C_j is that he/she did not learn C_i well. From the example given in Table 1, a concept effect graph is constructed as given in Fig. 3.

3.3. Learning diagnosis procedure

Table 2 displays a *test item relationship table* (TIRT) for a test sheet containing 10 test items ($Q_1,Q_2,Q_3,\ldots,Q_{10}$) on a learning unit for a subject involving the concepts illustrated in Fig. 3. Each value of $TIRT(Q_i,C_j)$, ranging from 0 to 5, represents the relationship between test item Q_i and the concept C_j. 0 indicates no relationship; 1–5 represent the intensity of the relationship; $SUM(C_j)$ denotes the total strength of concept C_j in the test sheet; $ERROR(C_j)$ is the total strength of the incorrect answers which are related to C_j; and $ER(C_j)=ERROR(C_j)/SUM(C_j)$ represents the ratio of incorrect answers to the total strength of concept C_j.

Assuming that the student fails to answer Q_3, Q_6, Q_7 and Q_9, as indicated in Table 2, then we have $ER(C_1)=1/6=0.16$, $ER(C_2)=0/5=0$, $ER(C_3)=3/5=0.6$, etc., indicating that the student failed to answer 16% of the test items related to C_1, 0% of the test items related to C_2, 60% of the test items related to C_3, and so on. The ER values are then assigned to each concept in the conceptual effect graph, as displayed in Fig. 4.

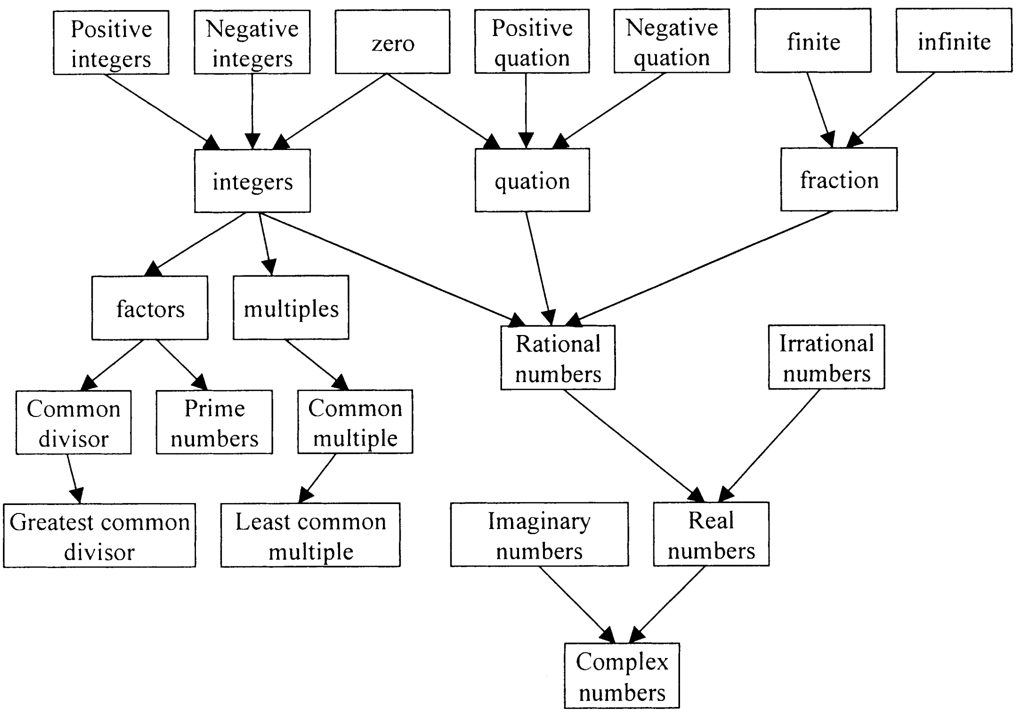

Fig. 2. Concept effect graph for the subject unit "numbers".

Table 1
Illustrative example of a concept effect table

Prerequisite (C_i)	C_j									
	C_1 (Zero)	C_2 (Positive integers)	C_3 (Addition)	C_4 (Odd)	C_5 (Even)	C_6 (Subtraction)	C_7 (Multiplication)	C_8 (Negative integers)	C_9 (Division)	C_{10} (Prime numbers)
C_1	0	0	0	0	0	1	0	0	0	0
C_2	0	0	1	1	1	0	0	0	0	0
C_3	0	0	0	0	0	1	1	0	0	0
C_4	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0
C_6	0	0	0	0	0	0	0	1	1	0
C_7	0	0	0	0	0	0	0	0	1	0
C_8	0	0	0	0	0	0	0	0	0	0
C_9	0	0	0	0	0	0	0	0	0	1
C_{10}	0	0	0	0	0	0	0	0	0	0
NP_j	0	0	1	1	1	2	1	1	2	1

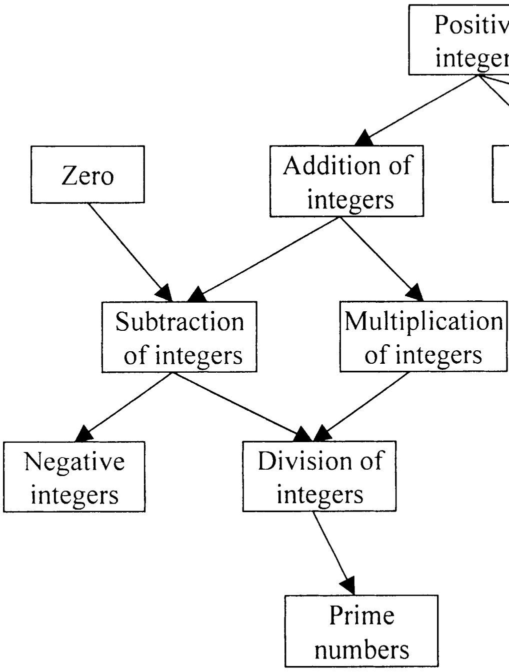

Fig. 3. Illustrative example of a concept effect graph.

Table 2
Illustrative example of a test item relationship table

Q_i	C_j									
	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}
Q_1	5	1	0	0	0	0	0	0	0	0
Q_2	0	4	2	0	0	0	0	0	0	0
Q_3	0	0	3	1	2	0	0	2	0	0
Q_4	0	0	0	5	0	0	0	0	0	0
Q_5	0	0	0	0	5	0	0	0	0	0
Q_6	1	0	0	0	0	4	0	2	0	0
Q_7	0	0	0	0	0	0	5	0	0	0
Q_8	0	0	0	0	0	0	0	0	1	0
Q_9	0	0	0	0	0	0	0	0	4	5
Q_{10}	0	0	0	0	0	2	0	1	0	0
SUM	6	5	5	6	7	6	6	5	5	5
ERROR	1	0	3	1	2	4	6	4	4	5
ER(C_j)	0.16	0	0.6	0.16	0.28	0.66	0.63	0.8	0.8	1.0

A threshold, θ, is used to indicate the acceptable error rate. When $\mathrm{ER}(C_j) < \theta$, the student is said to have learned concept C_j; otherwise, the student is said to have failed to learn concept C_j and thus the concept is added to the *To-Be-Enhanced learning path*. The value of θ can be determined by the following steps:

1. Calculate the lower bound of the error ratios for each concept. The lower bound for C_j, called $\mathrm{LB}(C_j)$, is determined by calculating the average error ratio of C_j for the students who get the bottom 50% of test scores.
2. The system calculates the difference in the student error ratio for concepts C_j and $\mathrm{LB}(C_j)$. Assume that the lower bounds of the concepts are $\mathrm{LB}(C_1) = 0.33$, $\mathrm{LB}(C_3) = 0.5$, $\mathrm{LB}(C_4) = 0.4$, $\mathrm{LB}(C_5) = 0.33$, $\mathrm{LB}(C_6) = 0.45$, $\mathrm{LB}(C_7) = 0.5$, $\mathrm{LB}(C_8) = 0.66$, $\mathrm{LB}(C_9) = 0.5$ and $\mathrm{LB}(C_{10}) = 0.66$, we have
$\mathrm{DIFF}(C_1) = 0.16 - 0.33 = -0.17$
$\mathrm{DIFF}(C_3) = 0.6 - 0.5 = 0.1$
$\mathrm{DIFF}(C_4) = 0.16 - 0.4 = -0.24$
$\mathrm{DIFF}(C_5) = 0.28 - 0.33 = -0.05$
$\mathrm{DIFF}(C_6) = 0.66 - 0.45 = 0.21$
$\mathrm{DIFF}(C_7) = 0.63 - 0.5 = 0.13$
$\mathrm{DIFF}(C_8) = 0.8 - 0.66 = 0.14$
$\mathrm{DIFF}(C_9) = 0.8 - 0.5 = 0.3$
$\mathrm{DIFF}(C_{10}) = 1.0 - 0.66 = 0.34$

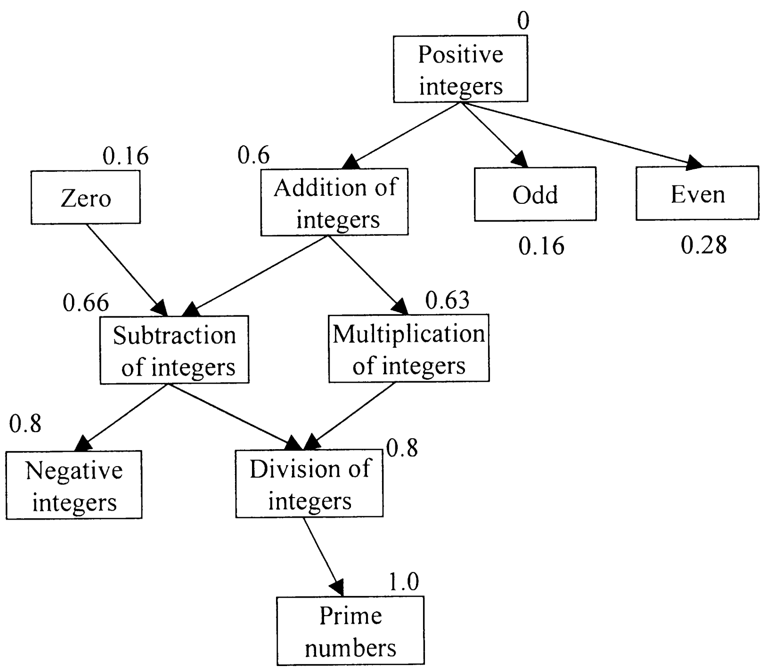

Fig. 4. Illustrative example of a concept effect graph with ER values.

3. The error ratios of C_1, C_4 and C_5 are lower than the corresponding lower bounds because their DIFF values are negative; therefore, the system suggests a value for θ by determining the minimum error ratio among C_3, C_6, C_7, $C_{8,}$ C_9 and C_{10}, i.e. $\theta = \text{MIN}\ (\text{ER}(C_j)) = 0.6$.

Consider the example given in Fig. 4, providing that θ is 0.6, the To-Be-Enhanced learning paths are as follows:

PATH1:	Addition→Subtraction→Negative integers
	Weight = Max(ER(C_3), ER(C_6), ER(C_8)) = 0.8
PATH2:	Addition→Subtraction→Division→Prime numbers
	Weight = Max(ER(C_3), ER(C_6), ER(C_9), ER(C_{10})) = 1.0
PATH3:	Addition→Multiplication→Division→Prime numbers
	Weight = Max(ER(C_3), ER(C_7), ER(C_9), ER(C_{10})) = 1.0

Among the To-Be-Enhanced learning paths, those with the maximum weight are defined as the critical learning paths, i.e. PATH2 and PATH3.

Clearly, the major problems in learning the subject unit originated from the misunderstanding of concepts C_3, C_6, C_7, C_8, C_9 and C_{10}. Furthermore, C_3 (addition of integers) is the root concept for the critical paths PATH2 and PATH3, and consequently students are asked to study C_3 in more detail before learning other concepts.

3.4. Fuzzy output for learning guidance

To make the learning guidance more understandable to the student, this study defines some fuzzy sets on ER(C_j), assuming that student understanding of a concept is determined by the ratio of incorrect answers they provide to test items related to that concept. Accordingly, Fig. 5 illustrates a fuzzy membership function related to learning status of concepts and ER(C_j).

Assuming that the ratio of incorrect answers provided by a student to test items related to concept C_1 is ER(C_1) = 0.16, then from Fig. 5, that student's learning status of concept C_1 is s follows:

Very well-learned→0.3
Well-learned→0.55
More or less well-learned→0.2

This study concludes that the learning status of the student for concept C_1 is "well-learned". Table 3 displays an illustrative example for the learning guidance depicted to the student.

4. Developing an intelligent testing and diagnostic system

The novel approach is used to implement a web-based intelligent testing and diagnostic system, ITES, on a Windows NT platform (see Fig. 6). The rules for constructing the test sheet are presented in the CLIPS format, a well-known expert system shell developed by NASA (Giarratano & Riley, 1989). Users can access this system via a WWW browser.

ITES comprises a student profile database, item bank, Java-based interface, testing and diagnostic unit, and fuzzy interface.

The student profile database contains the general information and learning status of each student. This database provides necessary information to the testing and diagnostic unit, which then invokes the fuzzy expert system to construct test sheets by analyzing the learning status of each student.

The Java-based user interface allows users to access the system via WWW browsers. Meanwhile, the user interface also controls both the registration and login processes. After successfully logging in, the user can adjust the parameters of the test items and take a test or can participate in the on-line discussions. An explanation of the test results is presented to guide users in further learning.

The testing and diagnostic unit is used to assign suitable test items to students and to record basic student information. By invoking the inference engine of the fuzzy expert system, a feasible test sheet can be constructed based on the parameters supplied and student records. Meanwhile, the fuzzy interface translates the user-requested test item parameters into fuzzy parameters. These fuzzy parameters, along with a set of fuzzy rules, are then forwarded to an expert system to start the inference process. Several tasks in the testing and diagnostic system are related to the fuzzy inference, such as the construction of the test sheets, the presentation of learning suggestions, and the determination of students' learning status.

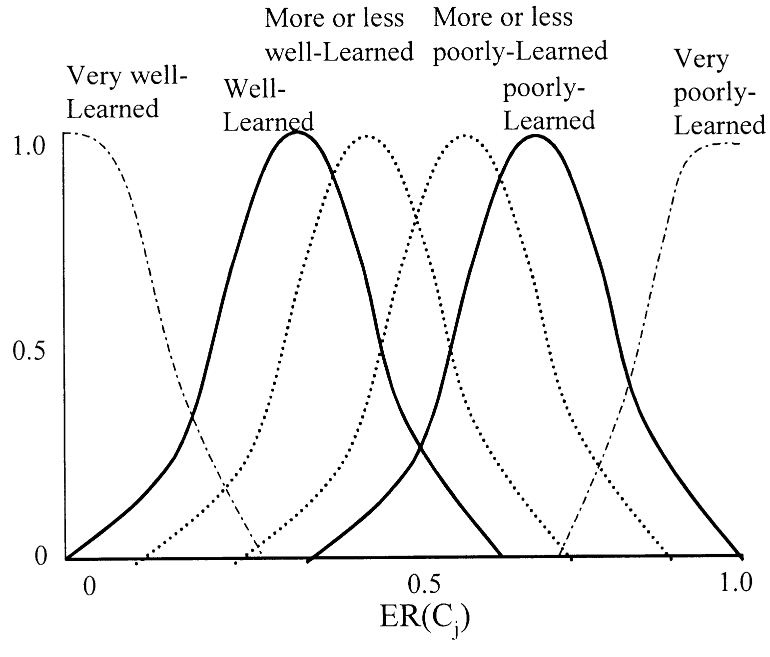

Fig. 5. Membership functions for learning status of concepts.

5. Administrations and evaluation

A graphical user interface is provided for constructing the conceptual effect graph, as shown in Fig. 7.

A user interface (see Fig. 8) is also provided to help the teacher to identify the relationships between subject concepts and test items. An on-line discussion window serves students who wish to take an examination for self-assessment. Students can either initiate a new discussion group, or join the discussion in existing groups. Students are allowed to interact on-line and to answer test items together. However, the discussion can be disabled if desired, for example when tests are being administered. Fig. 9 presents a mathematics related test item as an example.

Once students have submitted their answers, the server collects the answers from clients and saves them into five text files: question-id file, user answers file, correct solutions file, related concepts file, and the file containing weight of each concept in each of the questions. These five text files are then transmitted to the fuzzy expert system. After processing the inference flow, two text files are generated: a file for well-learned concepts and another for poorly-learned concepts. Based on these files, the server presents learning guidance to users, as shown in Fig. 10. Furthermore,

Table 3
Illustrative example of a learning guidance

Concept	Learning status of the concept
C_1 Zero	You have learned the concept well.
C_2 Positive integers	You have learned the concept very well.
C_3 Addition	It seems that you more or less misunderstood this concept.
C_4 Odd	You have learned the concept well.
C_5 Even	You have learned the concept well.
C_6 Subtraction	It seems that you misunderstood this concept.
C_7 Multiplication	It seems that you more or less misunderstood this concept.
C_8 Negative integers	It seems that you seriously misunderstood this concept.
C_9 Division	It seems that you seriously misunderstood this concept.
C_{10} Prime numbers	It seems that you seriously misunderstood this concept.

To-Be-Enhanced learning paths:
PATH1: Addition→Subtraction→Negative integers (0.8)
PATH2: Addition→Subtraction→Division→Prime numbers (1.0)
PATH3: Addition→Multiplication→Division→Prime numbers (1.0)

Critical learning path:
PATH2: Addition→Subtraction→Division→Prime numbers (1.0)
PATH3: Addition→Multiplication→Division→Prime numbers (1.0)

Comments for the student:
1. According to the diagnosis from the system, we found that you have misunderstood concepts "Subtraction", "Negative integers", "Division", "Multiplication" and "Prime numbers", which perhaps results from the misunderstanding of "Addition". In other words, the major learning problem of yours is the misunderstanding of concept "Addition", which affects the learning of other concepts.
2. Suggestion: enhance the study in "Addition→Subtraction→Division→Prime numbers" and "Addition→Multiplication→Division→Prime numbers" sequences.

the tutoring system arranges new learning paths and relevant homework for each student based on the learning guidance and the student profile.

To evaluate the efficacy of the novel approach, an experiment was conducted from September 2001 to December 2001 on the natural science course taught at an elementary school. Sixty K-6 students from two classes taught by the same teacher participated in the experiment, and were separated into two groups, A (control group) and B (experimental group), each containing 30 students. The students in *Group-A* (V1) received regular on-line tutoring and testing without learning guidance, while those in *Group-B* (V2) received the same on-line tutoring and testing, but with learning suggestions and relevant homework being supplied after each test. All 60 students were given three tests over the space of one semester (including a pre-test and two post-tests). The statistical results from applying SAS to analyze the tests are presented here (see Appendix).

5.1. Pre-test

Table 4 lists the t-test values for the pre-test results. The mean scores for the pre-test reveal that Group A performed better than Group B. Since the "Pr > F" value is 0.4079, the t value of "Equal" variances is adopted. That is, $|t| = 2.32 > t\alpha(29) = 1.699$, which implies that the performance of Groups A and B in the pre-test differs significantly. Therefore, we can conclude that Group A performed significantly better than Group B in the pre-test, conducted before performing the experiment.

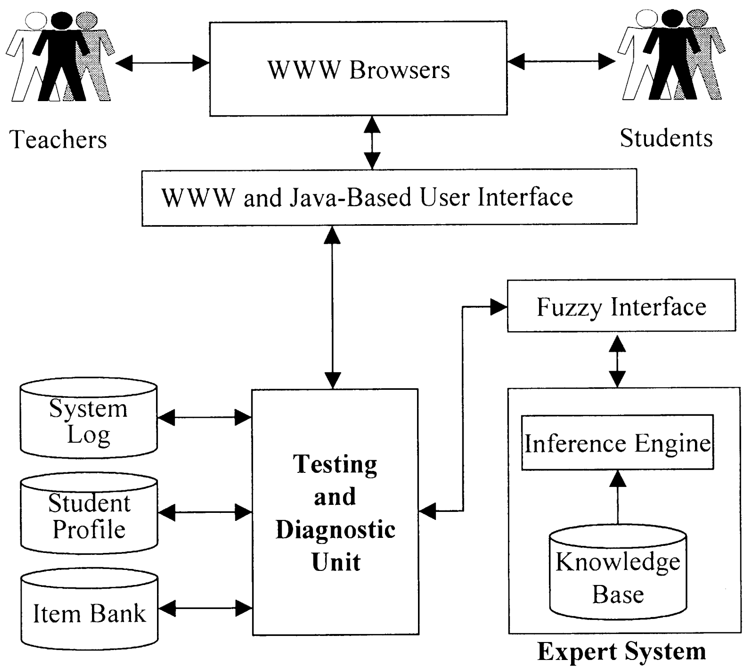

Fig. 6. Structure of ITES.

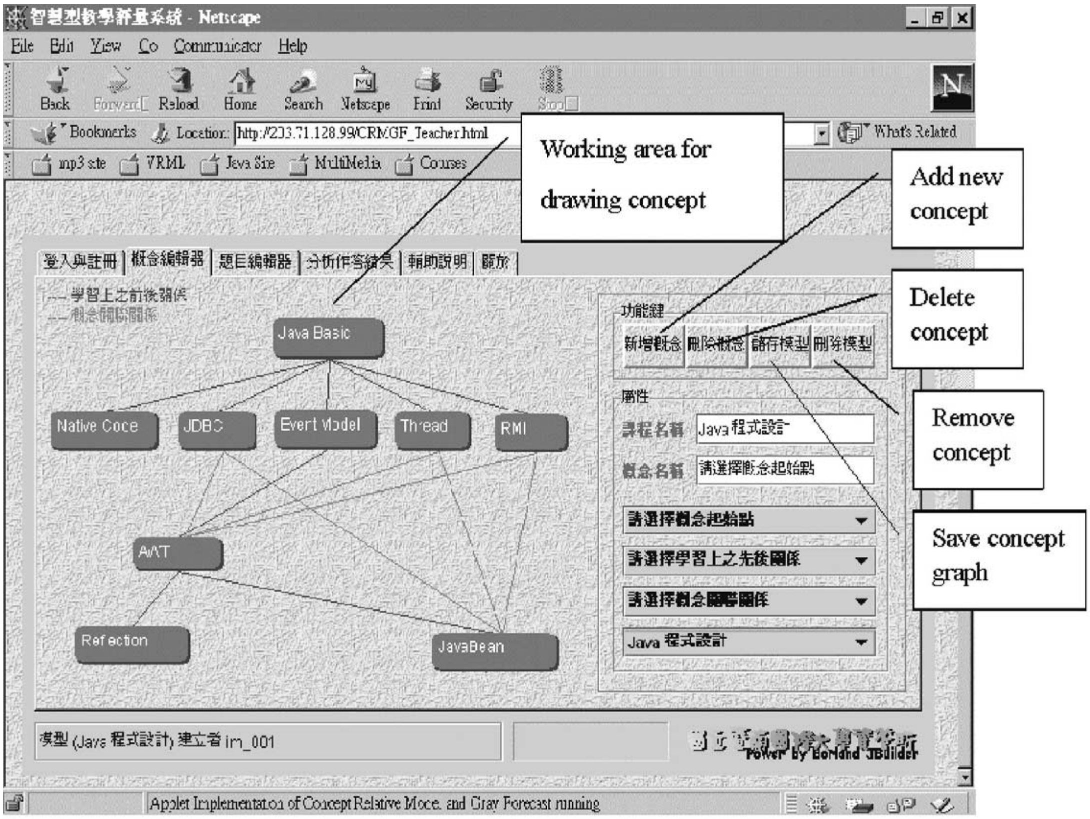

Fig. 7. Graphical user interface for constructing the concept effect table.

Table 4
t-Test of the pre-test results

Variable classes	*N*	Lower mean	Mean	Upper mean	Lower S.D.	S.D.	Upper S.D.	SE
Group A	30	80.775	83.067	85.358	4.8868	6.136	8.2487	1.1203
Group B	30	76.392	79.067	81.741	5.7045	7.1628	9.6291	1.307
Grade diff. (1–2)		0.5531	4	7.4469	5.6457	6.6692	8.1494	1.722

t-tests $\alpha = 0.05$, $t\alpha(29) = 1.699$

Variable	Method	Variances	df	*t* Value
Grade	Pooled	Equal	58	2.32
Grade	Satterthwaite	Unequal	56.7	2.32

Equality of variances

Variable	Method	Num df	Den df	*F* value	Pr > *F*
Grade	Folded *F*	29	29	1.36	0.4097

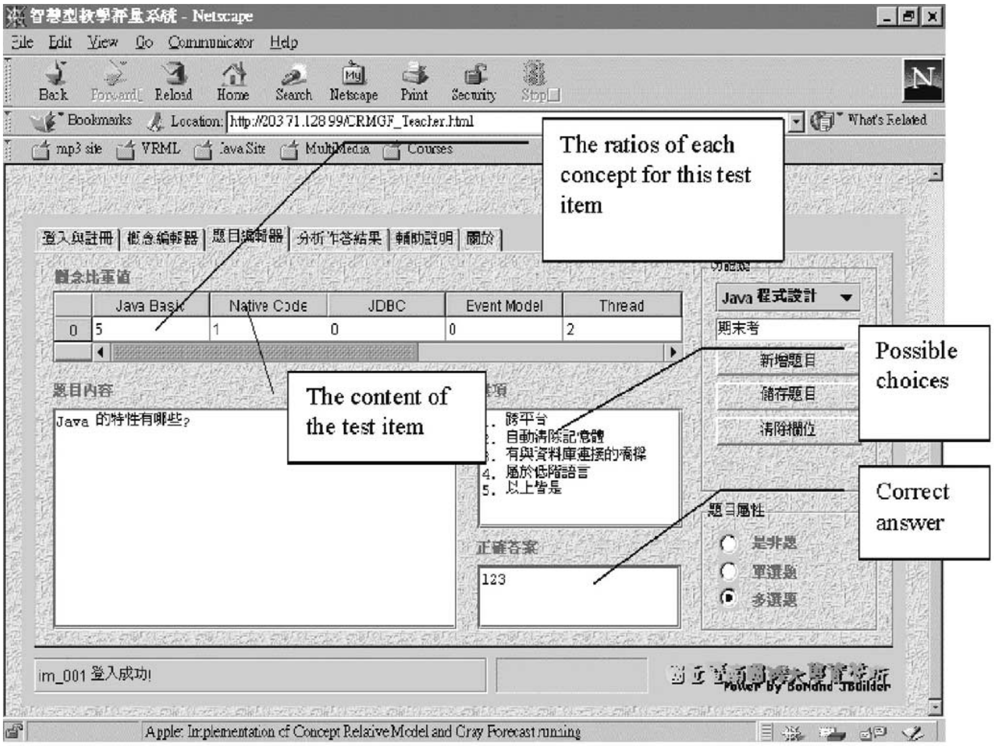

Fig. 8. User interface to construct the test item relationship table.

Table 5
t-Test of the post-test results

Variable classes	N	Lower mean	Mean	Upper mean	Lower S.D.	S.D.	Upper S.D.	SE
Group A	30	73.38	78.333	83.286	10.564	13.264	17.831	2.4217
Group B	30	82.149	85.7	89.251	7.5732	9.5092	12.783	1.7361
Grade diff. (1–2)		−13.33	−7.367	−1.402	9.7693	11.54	14.102	2.9797

t-tests $\alpha = 0.05$, $t\alpha(29) = 1.699$

Variable	Method	Variances	DF	t Value
GRADE	Pooled	Equal	58	−2.47
GRADE	Satterthwaite	Unequal	52.6	−2.47

Equality of variances

Variable	Method	Num df	Den df	F value	$\Pr > F$
Grade	Folded F	29	29	1.95	0.0782

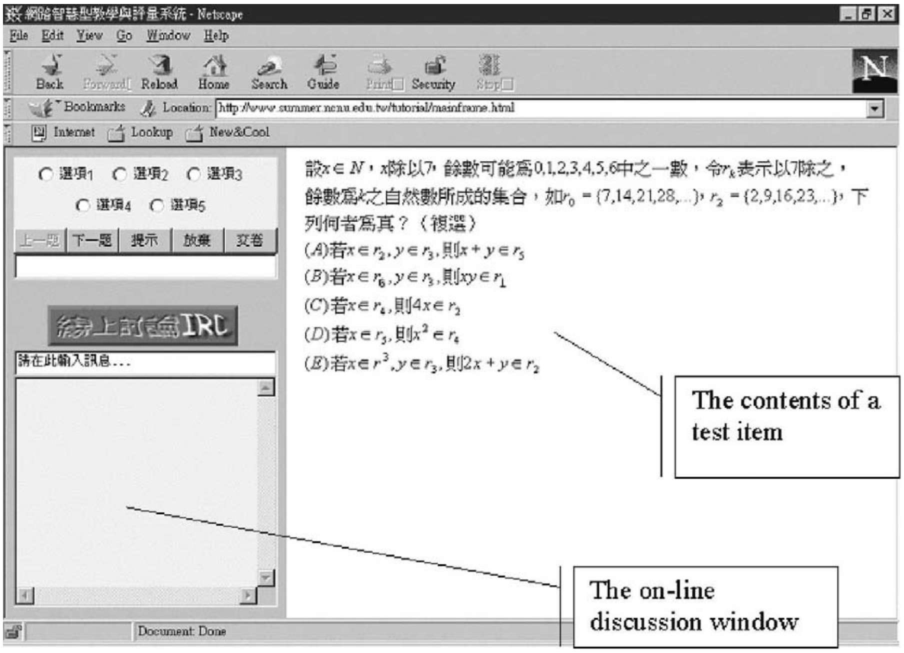

Fig. 9. Illustrative example of presenting a test item.

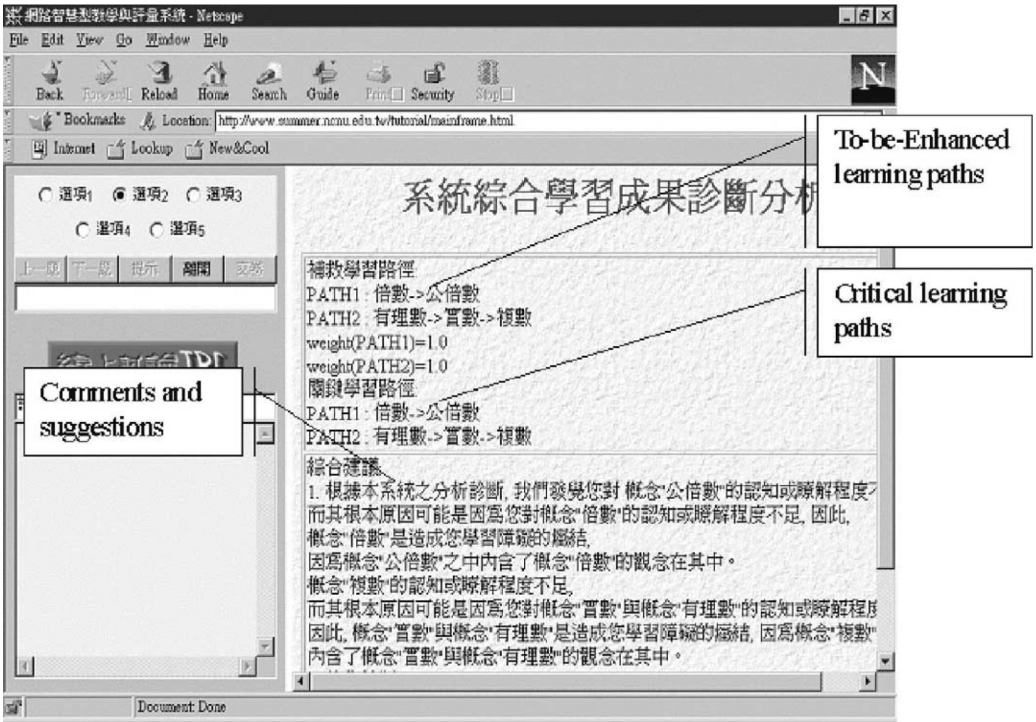

Fig. 10. Learning guidance presented by the system.

5.2. Post-test

Table 5 lists the t-test values for the post-test results. From the mean value of the post-test, Group B performed better than Group A. Since the "Pr > F" value is 0.0782 (not significant), the t-value of "Equal" variances is adopted, namely $|t| = 2.47 > t\alpha(29) = 1.699$, which implies a significant difference between the performance of Groups B and A in the post-test. Therefore, we can conclude that Group B achieved a significant improvement compared to Group A after receiving learning guidance via the novel tutoring system developed here.

6. Conclusions

This study proposes a conceptual map method for modeling the prerequisite relationships among concepts to be learned. Based on the proposed model, an intelligent tutoring system is implemented which can identify poorly-learned and well-learned concepts for individual students and provide appropriate individual learning guidance to enhance learning performance. To evaluate the efficacy of the novel approach, an on-line experiment involving 60 elementary school students was performed. The experimental results reveal that the experimental group, i.e. the group of students who received the learning guidance generated by the tutoring system, made significant progress compared with the control group. Consequently, we conclude that the novel approach helps students to improve their learning progress.

Although the experiment achieved positive results for a natural science course, it would be interesting to know whether the same approach would work, and how well, for various other kinds of courses, such as language courses, mathematics courses, science courses, engineering courses, and social science courses. Consequently, further investigations have been planned to apply the novel approach to on-line tutoring for different courses.

Another relevant topic is the development of a computer system to assist teachers in constructing item banks and concept effect relationships. One branch of our research group is attempting to develop a concept relationship construction system, which utilizes statistical theory and data mining technology to help teachers to identify relationships among concepts.

Developing an intelligent tutoring system clearly needs to consider various factors, including maintaining the item bank, constructing the concept relationships, generating feasible test sheets, implementing adaptive tutoring strategies and designing subject materials. Consequently, an extended project of ITES, the PTTD (Personalized Tutoring, Testing and Diagnosis Environment) project, was initiated in August 2001, and is scheduled to be finished in July 2004. The entire project is supported by National Science Council of Taiwan and aims to develop an integrated system that offers tutoring, testing and diagnosis functions. The conceptual map model proposed in this study will be employed to identify student learning status and to provide learning suggestions for individual students in the PTTD project.

Acknowledgements

The author would like to thank Miss Rei-Hsin Pon and Mr. Jin-Hon Liao for their assistance in performing the experiments and analyzing the experimental results.

Appendix. Test scores for natural science

Student ID	Pre-test	Post-test 1	Post-test 2	Student ID	Pre-test	Post-test 1	Post-test 2
A01	72	48	50	B01	90	96	96
A02	78	75	75	B02	72	69	71
A03	80	64	62	B03	82	83	95
A04	92	88	85	B04	86	80	86
A05	79	76	57	B05	82	85	85
A06	87	73	93	B06	81	65	84
A07	81	83	71	B07	83	90	93
A08	88	84	97	B08	76	88	91
A09	90	75	80	B09	88	93	85
A10	81	71	49	B10	86	70	86
A11	91	81	75	B11	71	81	71
A12	87	69	95	B12	85	84	93
A13	71	95	82	B13	74	64	54
A14	71	65	46	B14	80	79	73
A15	87	71	69	B15	69	75	63
A16	79	64	77	B16	86	85	89
A17	84	83	67	B17	85	87	98
A18	82	66	76	B18	67	79	80
A19	86	78	76	B19	73	79	88
A20	90	96	97	B20	70	97	85
A21	78	94	65	B21	86	99	91
A22	89	100	95	B22	73	97	98
A23	91	97	96	B23	90	99	95
A24	74	90	86	B24	79	92	97
A25	82	98	84	B25	67	93	82
A26	90	100	97	B26	80	99	93
A27	86	95	93	B27	81	72	93
A28	80	94	53	B28	69	78	92
A29	85	98	71	B29	75	93	95
A30	81	89	95	B30	86	96	99

References

Fan, J. P., Tina, K. M., & Shue, L. Y. (1996). Development of knowledge-based computer-assisted instruction system. *1996 International Conference Software Engineering: Education and Practice*, Dunedin, New Zealand.

Giarratano, J. R., & Riley, G. (1989). *Expert systems: principles and programming*. Boston: PWS-KENT Publishing.

Giraffa, L. M. M., Mora, M. da C., & Viccari, R. M. (1999). Modeling an interactive ITS using a MAS approach: from design to pedagogical evaluation. *Third International Conference on Computational Intelligence and Multimedia Applications*, *1999*, 153–158.

Gonzalez, A. J., & Ingraham, L. R. (1994). Automated exercise progression in simulation-based training, *IEEE Transactions on Systems, Man and Cybernetics*, *24*(6), 863–874.

Harp, S. A., Samad, T., & Villano, M. (1995). Modeling student knowledge with self-organizing feature maps. *IEEE Transactions on Systems, Man and Cybernetics*, *25*(5), 727–737.

Hopper, S. (1992). Cooperative learning and computer-based instruction. *Educational Technology Research & Development*, *40*(3), 21–38.

Hsu, C. S., Tu, S. F., Yeh, S. Y., Chu, C. Y., & Hwang G. J. (1997). Development of an intelligent testing and evaluation system on computer networks. *1997 National Computer Symposium, Taiwan*, pp. D129–134.

Hwang, G. J. (1998). A tutoring strategy supporting system for distance learning on computer networks. *IEEE Transactions on Education*, *41*(4), 1–19.

Hwang, G. J. (1999). Development of an intelligent testing and diagnostic system on computer networks. *Proceedings of the National Science Council of ROC*, *9*(1), 1–9.

Johnson, W. B., Neste, L. O., & Duncan, P. C. (1989). An authoring environment for intelligent tutoring systems. *IEEE International Conference on Systems, Man and Cybernetics*, *1989*, 761–765.

McAleese, R. (1994). A theoretical view on concept mapping. *ALT-J.*, *2*(1), 38–48.

McAleese, R. (1998). The knowledge arena as an extension to the concept map: reflections in action. *Interactive Learning Environments*, *6*, 1–22.

Ozdemir, B., & Alpaslan, F. (2000). N An intelligent tutoring system for student guidance in Web-based courses. *Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies*, *2*, 835–839.

Pugliesi, J. B., & Rezende, S. O. (1999). Intelligent hybrid system for a training and teaching environment. *Third International Conference on Computational Intelligence and Multimedia Applications, 1999*, pp. 148–152.

Rosic, M., Slavomir, Stankov, & Glavinic, V. (2000). Intelligent tutoring systems for asynchronous distance education. *10th Mediterranean Electrotechnical Conference, 2000*, pp. 111–114.

Rowe, N. C., & Galvin, T. P. (1998). An authoring system for intelligent procedural-skill tutors. *IEEE Intelligent Systems*, *13*(3), 61–69.

Salisbury, D. F. (1998). Effect drill and practice strategies. In David H. Jonassen (Ed.), *Instructional designs for microcomputer courseware* (pp. 103–124). New Jersey: Lawrence Erlbaum Associates.

Sun, C. T., & Chou, C. (1996). Experiencing CORAL: design and implementation of distance cooperative learning. *IEEE Transactions on Education*, *39*(3), 357–366.

Vasandani, V., & Govindaraj, T. (1994). Intelligent diagnostic problem solving tutor: an experimental evaluation. *IEEE Transactions on Systems, Man and Cybernetics*, *3*, 1739–1744.

Vasandani, V., & Govindaraj, T. (1995). Knowledge organization in intelligent tutoring systems for diagnostic problem solving in complex dynamic domains. *IEEE Transactions on Systems, Man and Cybernetics*, *25*(7), 1076–1096.

Vasandani, V., Govindaraj, T., & Mitchell, C. M. (1989). An intelligent tutor for diagnostic problem solving in complex dynamic systems. *IEEE International Conference on Systems, Man and Cybernetics*, *2*, 772–777.

Wainer, H. (1990). *Computerized adaptive testing: a primer*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Wong, L. H., Quek, C., & Looi, C. K. (1998). TAP: a software architecture for an inquiry dialogue-based tutoring system. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, *28*(3), 315–325.

Yoshikawa, A., Shintani, M., & Ohba, Y. (2000). Intelligent tutoring system for electric circuit exercising. *IEEE Transactions on Education*, *35*(3), 222–225.

Gwo-Jen Hwang was born on 16 April 1963, in Taiwan, Republic of China. In 1991, he received his PhD degree from the Department of Computer Science and Information Engineering at National Chiao Tung University in Taiwan. He is now an associate professor of National Chi-Nan University. His research interests include computer-assisted learning, computer-assisted testing, expert systems and mobile computing.