# Educational Hypermedia System *ALICE*: an Evaluation of Adaptive Features

ALENKA KAVČIČ, MARKO PRIVOŠNIK, MATIJA MAROLT, SAŠA DIVJAK
Faculty of Computer and Information Science
University of Ljubljana
Tržaška 25, 1000 Ljubljana
SLOVENIA

*Abstract:* - This paper presents a Web-based educational hypermedia system, called *ALICE*, and focuses on the evaluation of its adaptive features. The system builds on the concept domain model and includes the elements of knowledge uncertainty in the process of user modelling. The two models are used for adapting system's navigation support to an individual user; link annotation, direct guidance and link insertion are implemented in the system. The system has been evaluated in a real environment on the course Introduction to Java as a testing teaching domain. The results of the experiments have shown that the presented educational system and its adaptation techniques provide a valuable, easy-to-use tool, which positively affects user knowledge acquisition and thus leads to better learning results.

*Key-Words:* - Educational hypermedia, adaptive systems, adaptation technologies, user modelling, knowledge uncertainty, fuzzy set theory, evaluation.

## 1 Introduction

The role of computers in education has changed significantly in the last decade. Educational systems have exceeded passive learning systems and now play an active role in the learning process. The use of better and more efficient teaching and learning methods, combined with the power of information technologies, offers additional challenges and opportunities in distance learning, lifelong learning and e-learning in general.

With the growth and popularity of the Internet and the World Wide Web, the Web-based educational systems are putting on their value. They are becoming more and more attractive, because they are based on open standards, have low development and maintenance costs, support dynamic multimedia contents and can be viewed by free, yet powerful Web browsers. They are also quite easy to use, platform independent and offer various communication and cooperation opportunities as well as easier distant access.

The incorporation of more complex intelligent tutoring techniques enables educational systems to recognize individual users, their needs and consequently adapt the instructional sequence. Such systems are also able to adapt information and its presentation to each individual user, and dynamically support user's navigation through the hypermedia material. This ability to adapt to the individual user needs can significantly improve the teaching process as it has been shown that the best teaching method is individualized tutoring [2].

The paper focuses on evaluation of an adaptive educational system *ALICE*. First, the system is described together with the adaptive techniques used. The evaluation of developed system and its adaptive features follow. The results of this evaluation are presented and discussed.

## 2 Educational Hypermedia System *ALICE*

The educational hypermedia system called *ALICE* (Adaptive Link Insertion in Concept-based Educational system) was designed and developed at Faculty of Computer and Information Science, University of Ljubljana [4]. It is a Web-based educational system, designed for an arbitrary teaching domain (a tutoring system shell; the teaching content is added on top of this framework).

The system is able to adapt itself to each particular user, thus it has to be aware of the teaching domain, the individual users and their knowledge, and has to monitor their learning progress. Hence, the domain model and the user model are the most essential parts of the adaptive system, beside the adaptation model [7]. The domain model represents the teaching domain and its structure, and is a foundation for the user model and the adaptation model. The user model saves information on the user, reflects their progress and furthermore influences the adaptation of the system. The adaptation model takes care of the adaptive features of the system and affects the way the material is displayed to the user. The structure of the adaptive hypermedia system is shown in Fig. 1, where the arrows represent the described relations between the three models and the user. The developed system *ALICE* also follows the proposed structure.

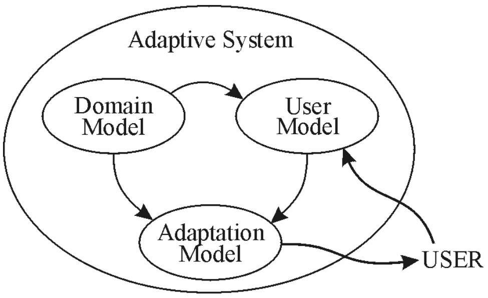


Fig. 1 Adaptive Hypermedia System Structure

*ALICE* is a fully Web-based system, which consists of the server and the client applications (see Fig. 2).

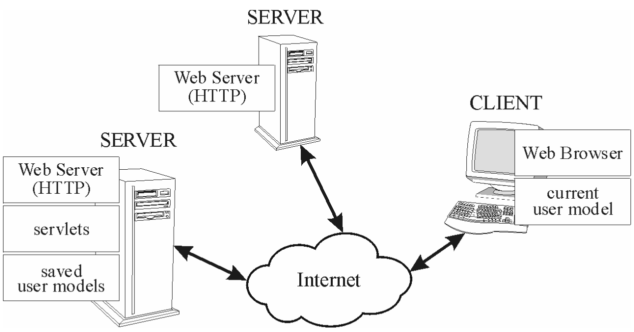


Fig. 2 Client/Server View on *ALICE*

The client's side application, which runs in the Web browser on the user's computer, displays teaching material in the browser, calculates, updates and maintains the current user model, and provides adaptive navigation support at the same time. The server application, implemented as Java servlets, provides login procedure, arranges pre-test results processing (for user model initialisation) and saves user model data on the server. The server's side application is used only at the beginning of the session for logging in and at the end for saving the updated user model. During the user's work with the system, the server only provides the requested Web pages (the teaching material), which can also reside on other Web servers in the Internet. This way, the server load and network traffic are significantly reduced, as client's resources are exploited for user model maintenance.

Next subsections describe some key components of the system.

## 2.1 Domain Knowledge Representation

The domain knowledge is logically partitioned into smaller parts or concepts. Hence, every domain can be represented by a finite set of domain concepts. The learning dependencies between the concepts are represented by ordered prerequisite relations: two domain concepts are related when the knowledge of the first concept is required for understanding the second one.

The prerequisite relation can be of two types [2]: essential or supportive. When two concepts are in the essential prerequisite relation, the first concept has to be mastered before the related concept, because it is crucial for understanding and learning the second concept. On the other hand, the supportive prerequisite concept just helps with understanding the related concept and is not necessarily learned first. Because the essential prerequisite relationship between two concepts can be present only to some extend, fuzzy relationship is used to model its presence [4].

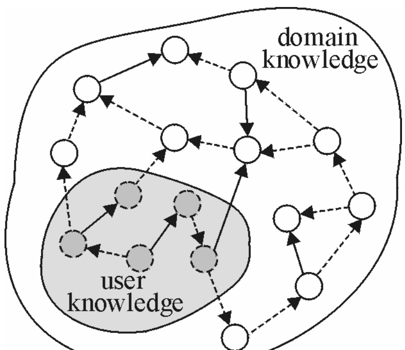


Fig. 3 Domain Concept Graph and User Model

The domain concepts, together with the relationships between them, can be represented by a directed acyclic graph. As the essential prerequisite relation is a fuzzy relation with different degrees of membership, the domain graph is a fuzzy structure too [4].

Such domain concept graph, which is used for modelling the domain, is illustrated in Fig. 3. Nodes in the graph represent the domain concepts, which are interconnected through the prerequisite relations. The two types of relations are represented by full arrows (crisp supportive) and dashed arrows (fuzzy essential) in the figure.

## 2.2 Fuzzy User Modelling

The user model collects various information about the current user and explains who is being taught [6]. It is essential for realizing system's personalization and adaptation.

Since the most important user property in educational systems is user knowledge, the user knowledge of domain concepts is modelled in our system. The description of user knowledge is usually very imprecise and vague, and includes a great deal of uncertainty. To deal with this uncertainty, a fuzzy set theory [8] was chosen for user model representation (fuzzy sets) and model updating (fuzzy rules) [4].

### 2.2.1 User knowledge representation

The user knowledge is basically described as an overlay over the domain knowledge (see Fig. 3), where the user knowledge is a subset of the domain knowledge. The user model is thus a subgraph of the domain concept graph, with additional description of each concept from the subgraph. The user knowledge of a concept is described by a triple of membership functions for fuzzy sets of unknown, known and learned concepts. Consequently, a fuzzy graph with fuzzy relations and fuzzy nodes is a base for user knowledge representation [4], as shown in Fig. 3, where fuzzy nodes/concepts in the graph are represented by dashed circles.

### 2.2.2 Knowledge determination and updating

While interacting with the system, users learn new concepts and their knowledge and understanding of the teaching domain increase. This is the main purpose of any educational system. Consequently, the user model has to be updated to reflect the changes in user knowledge.

After the first login into the system, the user is required to solve a short pre-test. The results of the pre-test are then used for user model initialisation, where each domain concept becomes either fully learned or remains completely unknown.

Test questions and visits to learning units are the main principles for gathering information about the user knowledge. When the user successfully solves the test corresponding to a certain learning unit (and its main concept), the membership value for a set of learned concepts is increased for the concept, regardless of the previous knowledge level of this particular concept. If the test questions are not answered satisfactory, the value does not change. Similarly, when the user visits a learning unit that explains a certain concept, the value of membership function for a set of known concepts for this concept is increased.

Because the domain concepts are interrelated, we can also infer knowledge values of some concepts (the knowledge of essential prerequisite concepts is inferred on the basis of demonstrated concept knowledge). Hence, after every change of concept knowledge values, an inferring mechanism (knowledge value propagation) is triggered, which updates the values of all essential prerequisite concepts. This propagation algorithm is based on six fuzzy rules and works recursively on all essential prerequisite concepts, until it reaches the basic concepts that have no prerequisites.

## 2.3 Teaching Material

Since our system is designed for an arbitrary teaching domain, the content of the teaching materials is not important for system realisation. The teaching material is composed of many learning units, each explaining a set of domain concepts. The learning units are Web documents residing on any Web server.

Each learning unit is indexed by domain concepts: a set of domain concepts is assigned to each unit, where one concept that best describes unit's content is set as a main concept for that unit. Each domain concept has to be a main concept in exactly one learning unit and vice versa.

The learning units typically have no direct links to other units, because the links are separated from the unit content. The necessary connections between the units are indirectly specified in the domain concept graph.

## 2.4 Adaptation Techniques

The adaptation of the system is accomplished through navigation support techniques [1]; direct guidance, colour link annotation and link insertion are implemented in the system.

The technique of adaptive link insertion [4] is a combination of link hiding and link sorting, where the system for each displayed unit dynamically adds groups of links to other relevant units (dynamic unit linking). The selection of links depends on the level of user knowledge, the current learning unit, the educational states of all units and the prerequisite relationships between the domain concepts. The selection of links, composed in this way, does not include all possible links from the current unit (hiding) and reorders the links by their suitability (sorting).

Users always select learning units by themselves; the system just supports them in their decisions. The unit can be selected from the table of contents (hierarchical navigation), from the lists of links to related learning units (relational navigation), with Back and Next buttons (linear navigation) or through the index of all domain concepts (concept-based navigation). In Fig. 4, a typical window of educational system *ALICE* is illustrated, where the currently displayed learning unit is on pointers in Java.

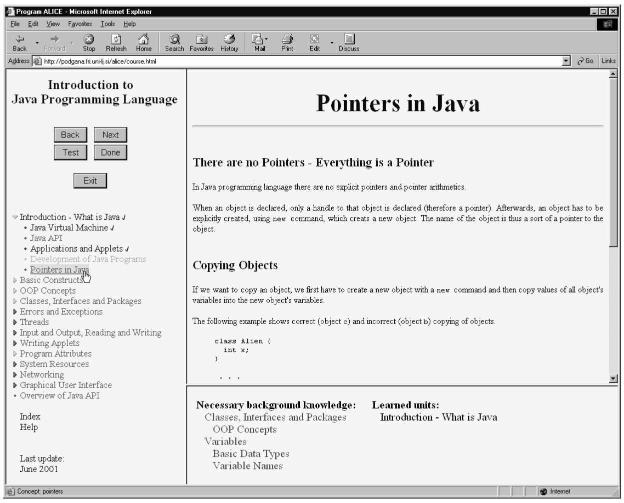


Fig. 4 Educational System *ALICE*

The window (in the Web browser) is divided into three parts. On the left side, the system provides navigation buttons, table of course contents, and links to index and help. At the bottom of the page, links to related learning units are displayed, which are determined by adaptive link insertion algorithm and are arranged into three groups (next suitable units, necessary background knowledge and already learned units). All links are colour annotated regarding their educational state: red link means the unit is not ready for learning and therefore not recommended, yellow stands for conditionally ready, green is recommended for learning, blue is for known units and black for already learned units. All studied units are checked (Done). Each unit also has corresponding test questions for assessing user understanding of the presented content (Test).

# 3 Different Versions of the System and Their Evaluation

The presented educational system *ALICE* was evaluated focusing on its adaptive features, usability and efficiency. Above all, we aimed to investigate the influence of adaptive system usage on learning ability.

The system versions used for the evaluation and the process of evaluation are described in the following sections.

## 3.1 Introduction to Java Course

For testing purposes, a teaching domain had to be chosen, prepared and integrated into the system. We decided to use Introduction to Java course, which was based upon an on-line tutorial on Java programming language [5].

The teaching domain was decomposed into 97 domain concepts and the prerequisite relations among the concepts were defined. Each of 97 learning units in 13 chapters (see the table of contents in Fig. 4) corresponds to one Web document, which describes the contents of this learning unit, and one set of test questions for checking user's knowledge. A pre-test was composed out of the selected questions. All learning units were indexed by the main domain concepts and a hierarchy of learning units for the table of contents was also established.

## 3.2 Preparation of Versions for Testing

The system was evaluated using two versions: fully functional adaptive system (version A) and limited adaptive system without colour annotation (version B). For both versions, the learning units (course content), the test questions, the general appearance and the functionality of the system stayed the same. Versions A and B differed only in the colour of the links.

## 3.3 Evaluation

The system was tested in a real environment by a group of first year Computer and Information Science students, all beginners in Java programming language. The students were randomly assigned to one of the two system versions A or B. First, the students solved a short pre-test. After about an hour of work with the system, they were asked to solve the post-test. At the end, they also answered the questionnaire. The collected post-test results and questionnaires were analysed in detail.

Regarding the system version and the use of system adaptive features, we divided the students into four groups: users of fully adaptive system that took advantage of system's adaptive features (A1), users of fully adaptive system that did not use adaptation (A2), users of limited system (without colour annotation) that used adaptation (B1) and users of limited system not using adaptation (B2).

# 4 Evaluation Results

The main objective of the evaluation was to find out, whether the use of adaptation affects the learning in any way. We focused primarily on the techniques of link insertion and link annotation. Besides, the general users' opinion on the system and its usability was also the point of our interest.

## 4.1 System Adaptation

In the evaluation of system's adaptive features, we analysed the collected post-test results in detail to establish the connection (if any) between the used adaptation technique and the post-test score. Since there were no significant differences in the pre-test results between the four groups, we supposed more or less the same level of previous knowledge of the teaching domain for all four groups.

The post-test results for each of the four groups are given in Table 1. The columns represent the group, the number of participants, the mean score (out of 17 points), its median and standard deviation. In average, groups A1 and B1 scored much higher (mean 12.97) comparing to groups A2 and B2 (mean 10.27). The difference in means is 2.7 or 16% of the maximum possible score. The ratio between both means is 1.26, meaning the groups A1 and B1 scored 26% better than the groups A2 and B2.

| Group | Num. | Mean | Median | Std.Dev. |
|---|---|---|---|---|
| A1 | 13 | 12.69 | 13 | 2.14 |
| A2 | 14 | 9.86 | 9 | 2.88 |
| B1 | 12 | 13.25 | 14 | 2.49 |
| B2 | 15 | 10.67 | 11 | 2.19 |
| **Total** | 54 | 11.52 | 12 | 2.75 |

Table 1 Post-test Results For the Four Groups

To prove the significance of this difference in group means, we carried out a one-way analysis of variance (ANOVA) on the post-test results. As the mean square for between groups (34.47) is much higher than the mean square for within groups (5.96) and the value of F ratio (5.78) is greater than the critical F value for 0.05 alpha level (2.76), the grouping variable does have an effect on post-test result. Hence, the difference between these four groups is indeed significant.

The F test showed that the groups are related to the post-test results, thus multiple comparison tests of significance had to be used to explore just which groups have the most to do with the relationship and estimate the size of the differences. Pair-wise comparison of groups (A1, A2, B1 and B2) compares all possible pairs of group means. For post-hoc pair-wise comparison we used Fisher's LSD method (least significant difference test) with individual 95% confidence limits (0.05 alpha level) [3]. The result of the multiple comparison tests showed that there were no significant differences in group pairs A1-B1 and A2-B2. On the other hand, the differences in group pairs A1-A2, A1-B2, B1-A2 and B1-B2 were identified statistically significant.

This means that the difference in post-test results of groups using the adaptive system features comparing to the groups not using them is indeed significant. In contrast, there is no statistically significant difference in post-test results between the groups that were offered colour annotation of links and the ones without it.

## 4.2 System Usability

After their work with the system, all students answered a questionnaire about the system and its usability. The answers, which reflect users subjective opinions, are summarized in Table 2.

As we can see from the answers (in Table 2), the system seemed easy enough to use, as well as its user interface and navigation. This is proved by the fact that more than half of the users (56%) needed less than five minutes to start working with the system and the majority (89%) needed less than 10 minutes to get used to the system. Most users enjoyed learning with the system (average score 4.0 out of 5) and would continue to use it in the future (94%). System's general usability grade (4.2) shows that the prototype was well designed. Most users were excited about the opportunity to simultaneously check their learning progress; the usability of Test button scored 4.6 points. Most users regard the colour annotation of links very useful (average score 4.4), although the analysis of post-test results did not show any particular influence on learning. Links to related units scored relatively low (only 3.6), but this can be the result of the fact that they were used only by half of the users (56%).

| **Questions about *ALICE*** | **A** | **B** | **Total** |
|---|---|---|---|
| Needed less than 10 min to start using the system (%) | 96 | 81 | 89 |
| Did you enjoy learning with the system (1 to 5) | 4.3 | 3.8 | 4.0 |
| Would you continue using the system (%) | 96 | 92 | 94 |
| User interface, navigation and system are easy to use (%) | 100 | 99 | 100 |
| Grade the system usability (1 to 5) | 4.4 | 4.0 | 4.2 |
| Grade the *Test* button usability (1 to 5) | 4.7 | 4.5 | 4.6 |
| Grade the colour link annotation usability (1 to 5) | 4.4 | – | 4.4 |
| Grade the adaptive link insertion usability (1 to 5) | 3.5 | 3.6 | 3.6 |
| Did you take advantage of inserted links to related units (%) | 58 | 54 | 56 |

Table 2 Answers to the Questionnaire on the Educational System *ALICE* (version A is full adaptive system and version B is adaptive system without colour annotation)

## 5 Conclusion

Adaptive educational hypermedia introduce a promising approach to Web-based tutoring. Beside their main advantages, such as distant accessibility, platform independence, easy preparation of learning material, and easy and cheap updating, maintenance and use, they also offer individualized teaching and support the learner in acquiring knowledge.

The results of the experiment showed that there is a significant difference, confirmed by ANOVA and pair-wise comparison, between the users that did take advantage of the adaptive features of the system and all the others. The former scored in average 26% better at the final examination. The use of colour annotation is not significant and has no direct effect on learning results, although the users liked it a lot and found it quite useful. In general, the users preferred adaptive systems, although only half of them used their adaptive features.

The presented educational system and its adaptation techniques thus provide an efficient, easy-to-use tool that positively influences user knowledge acquisition and consequently leads to better learning results.

*References:*


[1] P. Brusilovsky, Methods and Techniques of Adaptive Hypermedia, *Adaptive Hypertext and Hypermedia*, Kluwer Academic Publishers, 1998, pp. 1-43.

[2] R.M. Gagne, *Principles of Instructional Design*, Third Edition, Holt, Rinehart and Winston, New York, 1988.

[3] W. L. Hays, *Statistics*, 5th edition, Harcourt Brace College Publishers, Fort Worth, 1994.

[4] A. Kavčič, *Adaptation in Web-based Educational Hypermedia Considering the Uncertainty of User Knowledge*, Ph.D. Dissertation, Faculty of Computer and Information Science, University of Ljubljana, 2001.

[5] M. Marolt, M. Privošnik, A. Kavčič, *Introduction to Java Programming Language*, Teaching Material, Faculty of Computer and Information Science, Ljubljana, 2000.

[6] E. Wenger, *Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge*, Morgan Kaufmann Publishers, Inc., Los Altos, 1987.

[7] H. Wu, P. De Bra, A. Aerts, G.J. Houben, Adaptation Control in Adaptive Hypermedia Systems, *Adaptive Hypermedia and Adaptive Web-Based Systems, Proceedings of the International Conference AH 2000, Trento, Italy, August 2000, Lecture Notes in Computer Science*, Vol. 1892, Springer-Verlag, Berlin Heidelberg New York, 2000, pp. 250-259.

[8] L. A. Zadeh, Fuzzy Sets, *Information and Control*, Vol. 8, 1965.