An Adaptive Web-based Tutorial of Agrarian Economy

Cristina Carmona, Eduardo Guzmán, David Bueno, Ricardo Conejo

Departamento de Lenguajes y Ciencias de la Computación.
E.T.S.I. Informática. Universidad de Málaga. Apdo. 4114, 29080 Málaga. SPAIN.
{cristina,guzman,bueno,conejo}@lcc.uma.es

Abstract. *TEA* was originally conceived as a Computer Aided Educational System for teaching Agrarian Economy. This system was composed by web pages of instructional contents and web pages of test questions. The biggest flaw of the system is the lack of control over the pages seen by the student. This contributed to the disorientation of students during the instruction. Also, in test questions, students only could know if they had answered correctly to it, but these questions did not give them any feedback about their global level of knowledge in the topics of the tutorial. In this paper, *TEA* system has been improved making it adaptive. This has been achieved by using two tools: *SIGUE* and *SIETTE*. *SIGUE* is a web-based tool that converts non-adaptive courses into adaptive, and *SIETTE* is a web-based evaluation tool using adaptive tests. The functionalities of both tools have been integrated into a homogeneous adaptive *TEA*.

1 Introduction

There are many educational systems on the World Wide Web (WWW) that allow students to learn notions about different subjects. One of the main disadvantages of web-based systems is the disorientation during the navigation. In textbooks the order of the knowledge units is defined by the book's author, but in web courses this order is not so clear. The availability of multiple links in the same web page and the freedom to follow any of them, is not always an advantage for students. They can navigate without following a rational order in the pages of a course, and they can navigate in pages not related with the course. In these cases, it is necessary to have some control technique to guide the user during the study of a subject.

On the other hand, in most systems, there are no mechanisms to infer if students have assimilated the concepts studied in a page. Some mechanism of evaluation is required to assess the students' proficiency before letting the students access the pages of new concepts.

A characteristic of adaptive tutoring systems, is that they can estimate the knowledge level of a user in relation to a knowledge unit of a course. This information can be used to recommend which unit should be studied next. Adaptive tutoring systems provide the possibility of modifying the sequencing or presentation of the course for each user, according to their goals, current knowledge and/or preferences. The drawback is that, generally, they are difficult to implement due to the high development cost. So, most of existing tutoring systems on the web are not adaptive.

TEA [1] (the acronym stands for *Agrarian Economy Tutorial* in Spanish) was originally developed in the early years of the web, as a multimedia instructor system of Agrarian Economy. It is designed as a set of static web pages structured into chapters. Each chapter is also divided into sections. Special care was taken during the definition of the chunks of information in each page. At the end of each chapter, a set of questions is proposed to the students to evaluate their proficiency. *TEA* is available at http://www.lcc.uma.es/TEA.

The main goal of this paper is to demonstrate how a static instructor system can be turned into an adaptive course. To do this we will use of the *TEA* course. Two different tools have been used to this end: *SIGUE*, that allows building adaptive web pages from static ones; and *SIETTE*, which is a web-based system to construct adaptive tests.

In the next section, the *SIGUE* system is described, indicating the procedure followed to provide adaptive feature to *TEA*. In Section 3 a brief description of the main features of *SIETTE* is presented, as well as the adaptive mechanisms which it uses. This section ends with a description of *TEA* test in *SIETTE*. Section 4 explains how the integration of both systems has been accomplished. At last, contributions obtained from this work are approached.

2 Creating contents with SIGUE

SIGUE (http://www.lcc.uma.es/SIGUE) is an authoring tool that provide some adaptive features to existing static web-based courses. It is possible to create new courses reusing existing pages from different sources. This lets authors make courses by gathering the best information on the Web about a subject. For each concept of the course the author can choose how many URLs can be associated with this concept, indicating the kind of content (theory, examples or exercises) and the importance of this URL to the concept.

SIGUE provides a student's interface with commonly used adaptive tools, like annotated table of contents, colored buttons and progress bars. The student’s interface is implemented as a frame that serves as an additional navigation toolbar. Navigation on the main frame is also controlled depending on the specification of the teacher. For students that need guidance, *SIGUE* can recommend the next concept (and the best document within it) to go on, according to the student’s estimated knowledge. Another aspect of adaptivity is the multilingual interface of *SIGUE*, currently available in Spanish and English. A deeper description of *SIGUE* can be found in [11]

2.1 Making TEA adaptive

To create a course the first task to do is to define the domain model associated. The domain model is structured hierarchically. The root node contains the general information about the course. The nodes represent section and subsection. Leaves nodes have the information about the concepts being taught. *SIGUE* defines two binary relationships between nodes: "*belongs to*" and "*prerequisite of*". The first defines the hierarchy. If the author defines that the concept c1 *belongs to* c2, this means that c1 is a sub-concept of c2. One concept can have many sub-concepts but only one super-concept. The author can also define that c1 is *prerequisite of* c2. This means that to learn c2 it is necessary to study c1 first. The relationship of *prerequisite* defines a relationship of partial order. This gives an idea of the order in which concepts should be visited, which is the sequence of the curriculum.

The number of pages to describe a concept is unlimited, but there should be a main one that is shown the first time that a concept is visited. Authors must indicate the type of information that each page contains (theory, examples or exercises) and the difficulty (easy, normal or hard). The difficulty is used to order the pages within a concept. The way to present the pages will be; first, the main page followed by the rest, which are shown in order from the easiest to the hardest.

Once the domain model is created and all the concepts and relations have been defined, the author has to associate pages to concepts. The author can also define a glossary of terms for the course. A definition is associated to each term in the glossary. Optionally, a list of synonyms can refer back to the same definition. When a term defined in the glossary appears in any page of the course, a link is automatically inserted to show the definition of this term. This can be done in three ways: as a *hint*, this means that when student moves the mouse over the term, a textbox will show the definition; as an URL, the link will show a URL with the definition; or as an HTML page. In the last case, the author only has to write the HTML code that the student will see when he clicks on the term.

Courses in SIGUE can have different modes of operation. The author decides how adaptive the course will be and must set the mode of operation accordingly. There are four predefined modes: (a) Disable all the links in the pages shown to the student. In this mode the student will be able to do the course by accessing documents only through the links provided by the SIGUE navigation toolbar; (b) Leave all links of the pages, this lets users navigate freely even in pages not related to the course; (c) Enable only the links that give references to pages that belongs to the course. This option avoids distracting the user with pages not related to the course; (d) Fully adaptive. The links will be enabled according to the user model. Only those links corresponding to concepts that the user is ready to learn will be activated.

Fig. 1. The interface of *SIGUEAUTOR*.

To construct an adaptive course, it is necessary to define the prerequisite relationship of the concepts. This definition must be as complete as possible. This will be used to guide the student correctly in the sequence of concepts to visit, establishing a partial order for the learning process.

Fig. 1 shows the *SIGUE* authoring interface (*SIGUEAUTOR*). The client window is divided in two frames. On the left frame, the *SIGUEAUTOR* navigation toolbar is shown, the table of contents tree, the prerequisites table, the URL table, the access to the glossary, the mode of operation selector and the buttons to save the course and generate the XML file. On the right frame, the web browser shows the page visited. Each page is parsed by *SIGUEAUTOR* and its links are changed so that they will invoke *SIGUEAUTOR* again when they are clicked. Navigation is intercepted by *SIGUEAUTOR*, but this mechanism is transparent to the author. In this way, Authors can freely navigate and *SIGUEAUTOR* engine can always knows which page is loaded in the right frame. When an appropriate page is reached the page can be attached to a concept of the course by using the left frame .

2.2 Studying with TEA

Once the course has been developed with *SIGUEAUTOR*, it is available to students in *SIGUE*. While connected to the web through *SIGUE*, the student will see the hierarchical structure of concepts created by the author. All links and forms are modified according to the strategy selected by the author for the course. Some of them are eliminated and others are left (apparently) unchanged. The system will also include new references for the terms in the glossary, as defined by the author.

If the course is fully adaptive, a student model is created for each user. For each concept *SIGUE* shows two indicators. (a) The estimated background of the student to visit this page, that is, if he is prepared or not to read it, according to previous pages that have been visited. (b) The second is an indicator of the percentage of pages related to that concept that the student has already visited.

SIGUE does not have the ability to evaluate, it simply makes estimations of the knowledge based on the percentage of visited pages of each concept. So, if the percentage of URLs visited for a concept is less than the minimum the status of the concept is "empty"; if this percentage of visited pages is bigger than the maximum, the status is "full". The intermediary case is shown as "half-full". The status of a concept is associated to the percentage of pages the user has studied within a concept. So if the user has visited all the pages the system assumes that he has already learnt the concept. If it is "empty" the pages have not been visited yet and the system assumes that the concept is not known. This information is shown to the user by a progress bar that appears next to each concept.

The level of preparation necessary for a user to visit a concept is reflected by using colors in the nodes of the concept tree. Brusilovsky [9] justified the utilization of link annotation (equivalent to our color codes) through experiments done with their system *InterBook*. This system is also used in the lisp tutorial *ELM-ART* ([8] [10]). The use of this kind of annotation makes the user feel that there is some extra help in his interactions with the system, even if a "cognitive overhead may distract users from the content" [9].

In *SIGUE* the colors used are just green, red and orange. The color of the node is decided using the status of the prerequisites in this way:

- *Green*: A concept has this status when all the prerequisites are shown to have their status as "full". This means when all the prerequisites of this concept have already been covered, so the concept can be studied without difficulties.
- *Red*: This will be the color if at least one of the prerequisites of a concept is "empty". This means that the user has not studied one or some of the prerequisites. Red indicates that the user is not ready to study the concept.
- *Orange*: This indication will appear when no empty prerequisites exist. The student has begun to study all the prerequisites but has not finished all of them. Nodes with this color could be studied but finishing all the prerequisites first is recommended.

The student model is updated after each interaction, that is, every time the user visits a page. The concept tree is updated modifying the status and color of nodes accordingly.

The aim of *SIGUE* is to guide the student's navigation, and support it with adaptive annotation, but at the same time let the student move freely through the pages of the course. Access to any page is permitted for the user even if this is not recommended Another adaptive feature used to guide students is a simple planner that the student can use to get the best recommendation for the next document to view from the concept he is studying. (*SIGUE* means *continue* in Spanish.). This mechanism is designed for "two buttons users" and they are labeled BACK and NEXT.

Fig. 2 shows an example of the *SIGUE* student's interface for *TEA*. This course is adaptive, and in the image the progress bar for each concept and the colors that indicate the preparation of the student for each concept can be seen. On the right, the

HTML document associated with the concept and the activated link of recommended concepts and glossary terms (in this case activated by a hint).

Fig. 3 Example of the student's interface of SIGUE for TEA.

3 Creating tests with SIETTE

SIETTE (System of Intelligent Evaluation using Tests for Teleeducation) [2] is an adaptive web-based test generation system (http://www.lcc.uma.es/SIETTE. It has two main parts: (1) a set of *authoring tools*, that allow teachers to add and update contents in the knowledge base of the system. (2) A *virtual classroom* where students can make tests to evaluate their knowledge in certain subject.

The knowledge base is composed by the concept domain (*curriculum*), the specifications of the tests and the item pool. Concepts are structured hierarchically. The subject is the root node and the topics, and subtopics are its children. Items (that is questions) can be associated to any node in the hierarchy. An inheritance mechanism assumes that items of lower levels can also be used to evaluate their antecessors.

Tests are configured by teachers and they are defined by topics. To create a test, teachers must indicate which topics are going to be evaluated. When a student selects this test, the test generator will show him items from all topics involved in the test, or from any descendent subtopic of these topics. Therefore, there is not any direct relation between items and tests. This relation is established through topics.[11]

SIETTE is an adaptive evaluation tool. This means that the selection of the next question to pose to the students, the finalization decision of tests and the mechanism of estimation of the student's knowledge, are accomplished according to adaptive mechanisms. This kind of tests are known as *Computerized Adaptive Tests* (CAT) [3].

SIETTE, as well as the most of CAT systems, uses as an inference machine a psychometric theory called *Item Response Theory* (IRT) [4].

IRT is based on the hypothesis that the answer given to each item of the test, probabilistically depends on the knowledge level. As a result, conditional probabilities of the correct answer to the item by a student with a certain *knowledge level*, can be easily calculated for each item. This probability is expressed by means of a function $f : ?? ?, ? ? ?? \ ?0,1?$, named *Item Characteristic Curve* (ICC). The calculus of the ICC can be accomplished by several models. *SIETTE* uses a model of three parameters based on the logistic function [5].

Also, in IRT, the knowledge level of the student is estimated using the response to each item of the test. There are several methods to get this value. In *SIETTE* a Bayesian method [6] is used. In this method, the probability distribution of the student's knowledge level is calculated by the Bayes' rule. Also, it is assumed that the knowledge level can only take *K* discrete values (from *0* to *K-1*) because of the high computational cost of the calculus. Thanks to this consideration, the Bayesian method for the estimation of the student's knowledge level can be simplified to a vectorial product of ICC vectors with the *a priori* normalized density vector.

SIETTE does not need to establish priority relationships between topics. That is, there is no need to explicitly indicate if items from a topic must be posed to student before the items of other topics. If the characteristic curves of the items are well calibrated [7], thanks to the adaptive selection mechanisms, the test generator will show to the student the most suitable item according to his estimated knowledge level. This means that, if a student fails an item of difficulty *d*, the next item which the generator will pose him, will have a difficulty lower than *d*. Generally, the calibration process of items will make that items from the first topics of the curriculum, have lower difficulties than items of the latter topics.

The finalization criterion is configured by the teacher in each test. Through test editors, he must indicate a minimum and a maximum number of items. These values set bounds to the number of items that may be posed to the students. If a student has taken the maximum value of items, the final estimation process of his knowledge is forced. This ensures that test will finish, although the finalization criterion is not satisfied.

3.1 The test of TEA

In *SIETTE* a subject of Agrarian Economy has been defined. Its *curriculum* is composed by 14 topics. Each of this topics corresponds with one of the chapters of the *TEA* tutorial. There is a total of 84 items stored in the knowledge base. Each topic has approximately six of these items that are used to evaluate the knowledge of the student about this topic.

A test for each topic has been created. Using *SIGUE*, once the students has complete a chapter, they can directly access the corresponding tests in *SIETTE*, and evaluate their proficiency in that chapter. Additionally, a global test of the whole subject has been defined in *SIETTE* by using the inheritance mechanism.

All tests have been defined for twelve knowledge levels. Therefore, at the end, the system will give a qualification between 0 and 11. All items are multiple choice items, *i.e.*, items where students may either select only one response of a set of options, or not select anyone.

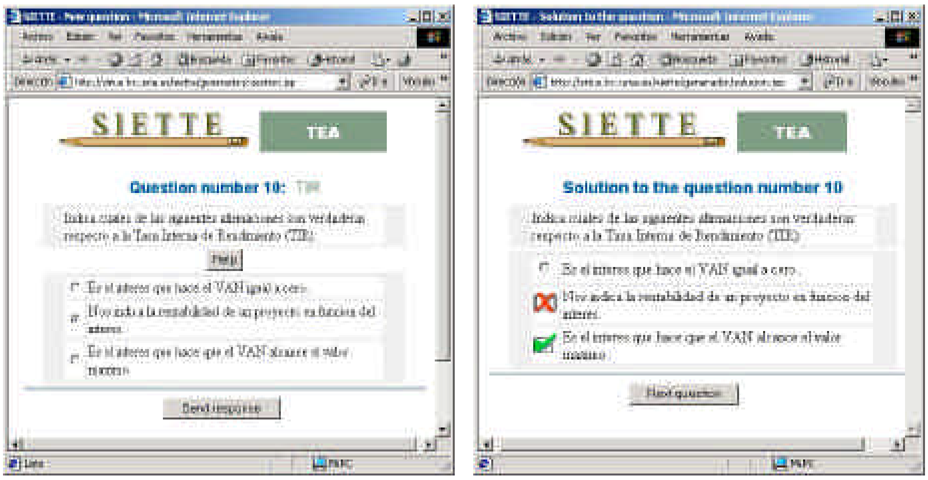

Fig. 4. An item posed by the test generator and its correction.

Fig. 4 shows an item presentation inside a test session (*a*), and its correction (*b*). Each item is composed by the title, the stem and the set of options. Once the student pushes the *Send Response* button, the system will show the correction of this item. Correct option is marked in green. If student selects a wrong option, it will be marked with a red cross. *SIETTE* can be configured to show the correct answer right after the student answer, at the end of the test or never. For educational purposes is advisable to correct the error as soon as it is detected. Different test could be make for practicing and for assessment, but currently the total number of question is not too high.

Fig. 5 shows the final qualification of the student in the test of the whole subject. As it can be seen, the system provides a detailed set qualifications for each topic. Statistics about the number of items posed to the student are given, as well as the number of items that have been successfully answered by the student. Note that the estimated knowledge level does not meet with the number of items successfully passed. This is due to the final qualification has been inferred by an adaptive criterion. At last, a pie chart with the percentage of topics posed from each topic, and the distribution curve of probabilities of that the student has each one of the knowledge levels, are presented too. Note that the value with the highest probability corresponds with the estimated knowledge level.

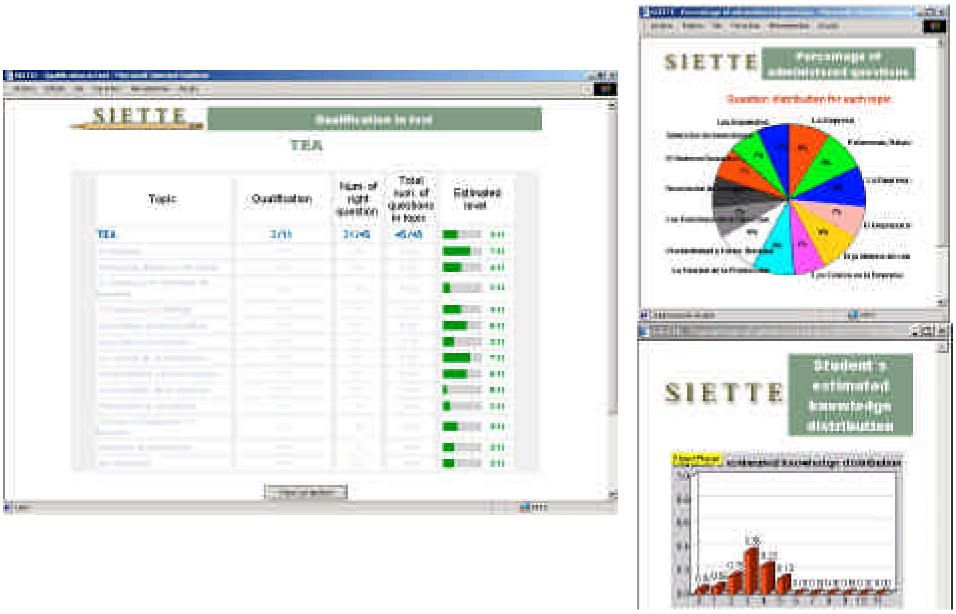

Fig. 5. Final qualification in the global test.

4 Integrating *SIGUE* and *SIETTE*

SIGUE and *SIETTE* have been integrated (it can be accesses through *SIGUE*, http://www.lcc.uma.es/SIGUE, selecting the *TEA* course) to make the access to the new adaptive version of TEA easy and homogeneous. As a result, students do not have to use separately both systems in order to have an adaptive instruction about Agrarian Economy.

In the navigation process *SIGUE* recommends the students which are the best pages to visit. .After each topic, the system will recommend to take a test in *SIETTE* to evaluate the proficiency in the topic. This test is also adaptive and is used with tutorial goals. However, at the end of the test the user has a feedback that will encourage him to continue with next topic or not.

Once the student has visited all the pages of the tutorial, the system will propose him to take a test of the whole subject. Through this test, students can check if they have assimilated correctly all the notions explained in the tutorial.

One of the advantages of the new design compared to the existing static course is that it can be easily updated. New content can be added, not only from *TEA* pages, but also from other pages related to the subject.

Figure 6 shows a hardcopy of a screen of both systems together. *SIGUE* has reach to a point where a test is recommended (see left frame). *SIETTE* has been invoked and the test is being done (right frame)

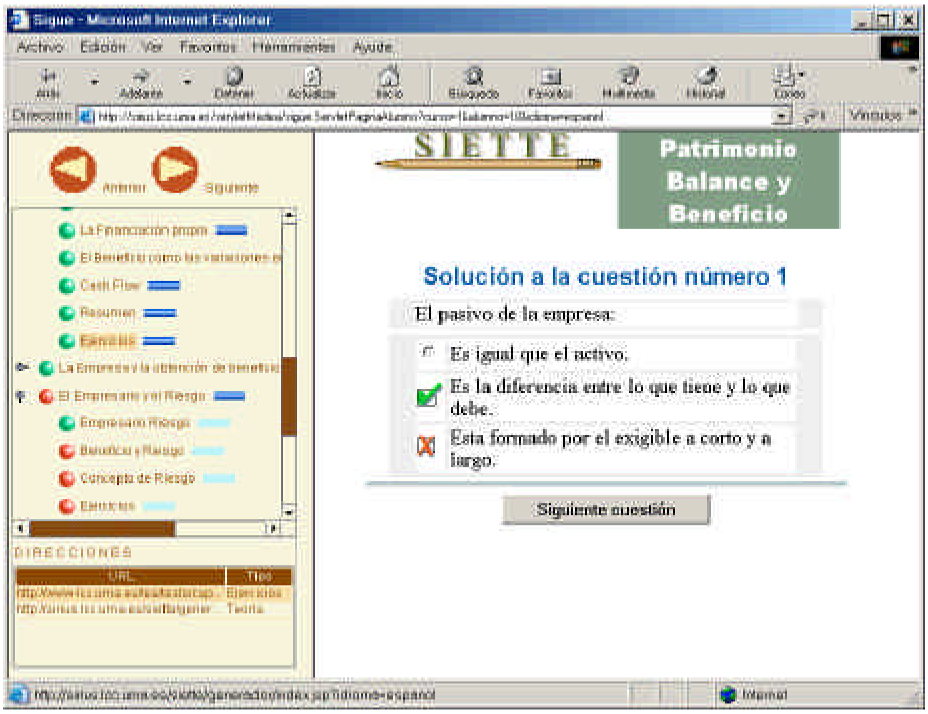

Fig. 6. Integration of *SIETTE* and *SIGUE*.

5 Conclusions

TEA was conceived as a Computer Aided Educational System for teaching Agrarian Economy. Even though it was initially designed to overcome the classical lacks of hypertext education, by using graphical and textual clues to improve the user orientation, the navigation through the course could not be controlled.

This educational system has been made adaptive. By using *SIGUE*, the navigation process of student is guided. The system annotates as the student goes through the hypertext the nodes visited and the estimated progress that the student should have at that point. This annotation is also used to guide the student upon request.

A set of adaptive tests have been include in TEA through *SIETTE*. There is a test for each topic, which allows students to evaluate their knowledge about that topic. In these tests, each item is selected in terms of the current estimated student's knowledge level.

This work has mainly supposed the integration of two web-based adaptive systems. One system to guide the student in the learning of concepts, and other system to evaluate his proficiency in these concepts. By means of this integration, students can access to both functionalities through the same web system. The student does not need to know that he is accessing two different systems. This makes easier the instruction process.

The drawback of the system is that there is no deeply integration between *SIGUE* estimations of the student's knowledge level and *SIETTE* assessment of student's knowledge level. *SIGUE* student model is basically a model of pages visited, and if the test is done, it assumes that the corresponding instruction has been completed.

SIGUE and *SIETTE* can also be integrated as components of *MEDEA* [12]. *MEDEA* is an open system to develop Intelligent Tutoring Systems. It is composed of independent educative modules coordinates by a core that controls the instruction of the student. *MEDEA* provide a communication protocol with their components that allows a deeper integration.

References

1. Belmonte, M., Berbel, J., Conejo, R. (1996). TEA: An Agrarian Economy Instructor System. *Third International Conference on Computer Aided Learning and Instruction in Science and Engineering, CALISCE'96*, San Sebastián. 322-330.
2. Ríos, A., Millán, E., Trella, M., Pérez-de-la-Cruz, J.L., Conejo, R. (1999) Internet Based Evaluation System. *Artificial Intelligence in Education:*, IOS Press, Amsterdam. 387-394.
3. Wainer, H. (ed.). (1990). *Computerized Adaptive Testing: a Primer*. Hillsdale, NJ: Lawrence Erlbaum Associates.
4. Lord, F. M. (1980). *Applications of item response theory to practical testing problems*. Hillsdale, NJ: Lawrence Erlbaum Associates.
5. Birnbaum, A. (1968). *Some latent trait models and their use in inferring an examinee's mental ability*. In Lord, F. M. & Novick, M.R. (ed.) *Statistical theories of mental test scores*. Reading, MA: Addison-Wesley.
6. Owen, R. J. (1975). A Bayesian sequential procedures for quantal response in the context of adaptive mental testing. *Journal of the American Statistical Association 70*, 351-356.
7. Conejo, R., Millán, E., Pérez-de-la-Cruz, J. L. & Trella, M. (2000). An Empirical approach to on-line learning in SIETTE. In *Proceedings of the ITS 2000*, Montreal. Springer-Verlag.
8. Brusilovsky, P., Schwarz, E., & Weber, G.(1996). ELM-ART: An intelligent tutoring system on World Wide Web. In Frasson, C., Gauthier, G., & Lesgold, A. (Eds.), *Intelligent Tutoring Systems*, Springer Verlag, Berlin (1996) 261-269.
9. Brusilovsky, P. (1998). *A Study of User Model Based Link Annotation in Educational Hypermedia, Journal of Universal Computer Science*, vol4, n-4,429-448
10. Weber, G., & Specht, M. (1997), User modeling and adaptive navigation support in WWW-based tutoring systems. In Jameson, A., Paris,C., & Tasso, C. (Eds.), *User Modeling*, Springer-Verlag, Wien 289-300.
11. Guzman,E. Conejo, R. :Simultaneous evaluation of multiple topics in SIETTE. En *Procdeedings ITS2002*. Springer-Verlag (2002) (to appear)
12. Trella, M., Conejo, R., Bueno, D., Guzmán, E.: (2002). An autonomous component architecture to develop WWW-ITS. In Brusilovsky, P., Henze, N., Millán, E. (eds.) *Proceedings of the Workshops on Adaptive Systems for Web-Based Education.* Malaga (this volume).