Authoring and Delivering Adaptive Web-Based Textbooks Using WEAR

Maria Moundridou
Department of Informatics,
University of Piraeus, Greece
mariam@unipi.gr

Maria Virvou
Department of Informatics,
University of Piraeus, Greece
mvirvou@unipi.gr

Abstract

In this paper, we describe the adaptive courseware authoring capabilities of WEAR, an ITS authoring tool for Algebra-related domains. The system allows the authoring of electronic textbooks and delivers them over the WWW. Then learners are offered navigation support adapted to their individual needs and knowledge. The domain knowledge and the information kept in learner models are used in WEAR to provide adaptive navigation to students and also to support the instructional designers in the authoring process. An instructor model incorporated in WEAR's architecture contains additional information that is exploited by the system to achieve the goal of authoring support.

1. Introduction

What best describes Adaptive Hypermedia Systems (AHS) is their ability to adapt the interaction with each individual user. Adaptation is based on the model they maintain of each user's goals, preferences and knowledge. Education is a popular application area for AHS since previous knowledge on the subject being taught, rate of progress, learning goals and other characteristics may vary a lot among learners. This is even more the case with Web-based education, which aims at reaching a much more heterogeneous group of learners. This challenging goal urged in the recent years a number of research groups to engage in research on adaptive Web-based educational systems. As a result, quite a lot of Web-based adaptive hypermedia systems exist, making use of various methods and techniques.

Brusilovsky in [2] provides a detailed review of adaptive hypermedia methods, techniques and systems and distinguishes between two main adaptive hypermedia technologies: (i) *adaptive presentation*, the case where adaptation is performed at content level and (ii) *adaptive navigation support*, which is performed at link level. Both these technologies have been evaluated and the results offer strong evidence that their use in an educational AHS can have a positive effect on students' learning and comprehension of the domain (e.g. [4], [9], [1]).

However, the development of a Web-based course is a quite time consuming task that cannot be carried out by the instructor or the author of the teaching material but rather requires the involvement of programmers and other experts. A way to overcome these problems is to develop a Web-based course using an authoring tool. Among the available tools of this kind, there are quite a few that support not only course creation and delivery but also other functions and utilities such as bulletin boards, course management, chatting etc. ([7], [14]). Unfortunately, the courses created with these powerful, commercial tools do not adapt in any way to the individual learner and are rather static. On the other side, there are tools originating from research efforts that support the development and delivery of adaptive Web-based courses. Among them, we cite Interbook [5], AHA [6] and MetaLinks [10].

In this paper, we will describe the adaptive courseware authoring capabilities of a tool we are developing and is called WEAR (WEb based authoring tool for Algebra Related domains). WEAR is a Web-based Intelligent Tutoring System authoring tool, which is mainly concerned with problem construction and solving in Algebra-related domains [12]. However, the adaptive textbook authoring facility it offers could also be utilised for the creation of courseware in other domains too, even if they are not Algebra-related.

In particular, WEAR allows the authoring of electronic textbooks by instructors and delivers them over the WWW to learners. These textbooks offer navigation support to students, adapted to their individual needs and knowledge. To achieve these aims, WEAR is based on three models: the domain model (representing knowledge about the domain of the subject matter), the learner model (representing knowledge about the individual learner) and the instructor model (representing knowledge about the instructor). Using information derived from these models the system can provide individualised support to instructors concerning the authoring of the course. The remaining of this paper is organised as follows: Section 2 describes the models of the system; in section 3 we present how adaptive navigation is performed and in

section 4 we describe the course authoring process and the support provided to designers with regard to adaptive navigation.

2. Models of WEAR

The *domain model* containing knowledge about the subject matter is structured as a network of hierarchically organised topics (textbook sections). Links between nodes of that network represent relationships between topics. At the moment, two types of relationship are used: *is_prerequisite_of*, to describe a topic a learner should know before accessing the more advanced one, and *is_related_to*, to describe that these two topics are in some way related to each other. Each topic has an associated *difficulty level* ranging from 1 (very easy) to 5 (very difficult). Finally, problems and/or tests examining the knowledge that must be acquired by studying a particular topic are associated with it; these associations are also part of the domain knowledge.

For each topic contained in the domain model, the individual *learner's model* stores two attribute-value pairs. These are: (i) *read* (true or false), indicates if this topic has been visited by the student, (ii) *knowledge weight* (ranging from 0 to 1), is an estimation of the student's knowledge level on this topic; it is calculated taking into account both the student's performance in solving the problems associated with this topic (if such problems or tests exist) and also the value of the *read* attribute.

The part of the learner model that is used for adaptive navigation is a combination of a stereotype and an overlay student model, similarly with other systems, such as [8]. The stereotype model (formed either directly by the instructor or after a preliminary test posed to the student) classifies initially the student according to his/her knowledge of the domain. As a result of this, each student is assigned to a stereotype (novice, beginner, intermediate or expert). The stereotype model also defines initial values for the overlay student model described above, taking into account each topic's *difficulty level*. If for example the stereotype model indicates that a student is "intermediate", then the initial value of the attribute *knowledge weight* will be 1 for all topics with difficulty level 3 or lower. The underlying assumption in this is that a student considered "intermediate" probably knows every topic which is not rated as difficult or very difficult.

Each time a student visits a topic, solves a problem or does a test, his/her user model is modified to reflect his/her current knowledge state in the domain being taught.

WEAR -unlike other tools- incorporates an instructor modelling component in its architecture [13]. With regard to the adaptive textbook authoring, the *instructor model* mainly holds information obtained explicitly by the instructor. Such information is the instructor's long-term preference concerning the difficulty of the course. The instructor is also asked to specify how the students' level of knowledge will be calculated. For example, an instructor may state that s/he wishes the reading of a topic by a student to be given a weight of 20% and the rest 80% of the knowledge level to be obtained from student's scores in problems and tests.

3. Adaptive navigation support

Brusilovsky in [2] describes several methods of adaptive navigation support, such as adaptive link sorting, annotation, or hiding, map adaptation and direct guidance. WEAR at the moment provides adaptive navigation support through adaptive link annotation. In particular, using the information stored in each learner's model and the domain knowledge, WEAR generates a table of contents consisting of links to every topic of the domain. These links are annotated in order to inform students about the educational appropriateness of the topic behind them. In that way, five different states of links can be distinguished in a Table of Contents (TOC) generated by WEAR, as shown in Table 1. To annotate different states of links WEAR uses different icons. For example, a checkmark next to a link means that this topic's state is "Visited and Known", whereas a "no-entrance" icon implies that this topic is "Not Ready" to be visited yet.

When a student visits a topic, s/he is offered the choice of seeing a list of related links. These links are also annotated in the same way as the links in the Table of Contents.

Table 1. Topic states

State	Rules
Visited but Not Known	If this topic is visited but its knowledge weight is lower than a threshold
Visited and Known	If this topic is visited and its knowledge weight is higher than a threshold
Not Ready	If this topic is either not visited or Visited but Not Known AND there is at least one prerequisite topic that is either not visited or Visited but Not Known
Ready and Highly Recommended	If this topic is Visited but Not Known and all prerequisite topics are Visited and Known
Ready and Recommended	If this topic is not visited and all prerequisite topics are Visited and Known

4. Authoring for adaptive navigation

Authoring an ITS with WEAR involves creating and structuring the teaching material, constructing problems

and tests and managing the student records. In the subsequent sections we will deal with the former (the textbook construction) since this is mostly related to adaptive navigation support.

4.1. Authoring procedure

Although most of the existing authoring tools for adaptive educational textbooks approach the adaptivity issue in quite similar ways, they differ a lot in the authoring process they impose to their users (authors). For example, in Interbook [5] the author should provide a specially structured, annotated MS-Word file. In AHA [6] the author should write annotated HTML files. MetaLinks [10] on the other hand, provides a GUI interface for authoring all aspects of the electronic textbook.

In WEAR, we address authoring in a way that in its first steps resembles the simple one adopted by commercial tools like WebCT [7]. In particular, the authoring procedure is the following: The author should prepare HTML files for the topics that would be contained in the electronic textbook. This is very trivial since there are tools both for creating HTML files and for generating HTML from other document formats. The next step is to use WEAR's facilities for uploading these files to the WEAR server. For each uploaded file the author must also specify a title, a difficulty level and the position that it should have in the topics hierarchy. Finally, the author must edit the is_prerequisite_of and is_related_to relationships between topics. To perform this, the author is presented with the hierarchy of topics and s/he should write in two columns next to each topic the section numbers of its prerequisite and related topics.

The author may also create multiple choice tests or problems and associate them with the appropriate topics. S/he may also create a preliminary test to be set to students in order to classify them in a stereotype and initialise their user model. In that case, the author should state which ranges in scores obtained from the preliminary test correspond to which stereotype. Alternatively, the author could manually define for each student of the virtual class the stereotype s/he belongs.

4.2. Authoring support

Some problems linked to the authoring of adaptive textbooks by instructors through the use of authoring tools are the following:

(i) Instructors may face several difficulties during the design process (e.g. they may not be sure about the structure their course should have), (ii) they may provide inconsistent information to the tool that may lead to the generation of adaptive textbooks with problematic behaviour (e.g. they may define prerequisite relationships in such a way that some sections of the textbook would be unreachable or never recommended), (iii) in order to be domain independent and generic, most authoring tools embody predefined pedagogical rules which cannot even be configured; instructors are obliged to accept them even if these rules contradict the way they perceive instruction.

To overcome some of these problems, several approaches were proposed. For example, Wu et al. in [15] describe support tools that help authors create usable and consistent adaptive hypermedia applications. Brusilovsky in [3] introduces a concept-based course maintenance system which can check the consistency and quality of a course at any moment of its life and assist the course developer in some routine operations. A more sophisticated approach is presented in [11] by Nkambou et al.: In order to provide designers with support that focuses on the expertise for building courses, they propose to use an expert-based assistant integrated with the authoring environment. The expert system reasons on a constraint base that contains constraints on curriculum and course design that come from different instructional design theories. In that way, the expert system validates curriculums and courses produced with the authoring tool and advises the instructional designer accordingly.

In WEAR, instructors are assisted in authoring a course along various dimensions. First of all, when building the course they are provided with tools that verify the course's consistency and report possible problems or errors (such as the case when the prerequisite relationships impose that a topic indirectly requires the knowledge of itself). To offer more intelligent and individualised help to instructors WEAR relies on the information provided by an instructor modelling component that it embodies. The information of this model and the learner model is used by WEAR to support instructors in the authoring process in the following ways:

Instructors are offered the choice to see what other instructors have done. The information that is presented to the user in that case, is the structure of a similar course (in terms of the domain to which it belongs and in terms of the difficulty level assigned to it by its author) created by another instructor. In particular, the instructor may see an enriched TOC presenting not only the topic hierarchy but also the prerequisite and is_related relationships between topics. In that way, instructors who may be novice as course designers could be assisted by more experienced peers who have previously used WEAR.

While students are working on the course, the system collects evidence to build reports and offer advice that may be of interest to the instructor. If most students are not doing well and the instructor's goal (as recorded in his/her user model) is to offer an easy course, then s/he is notified of the inconsistency. Furthermore, WEAR also performs more thorough checks: for instance, if the majority of students fail to comprehend a specific topic

(indicated by low scores in the corresponding tests), then the instructor is informed and given some suggestions concerning this situation (e.g. the underlying reason for the students' failure may be the misplacement of the specific topic in the curriculum, or otherwise it may be that the test was too difficult). By receiving feedback concerning the efficiency of the course they constructed, instructors can redesign it; multiple iterations of this process may lead to the construction of optimal courseware for their class.

5. Conclusions

In this paper we described WEAR, an authoring tool for adaptive Web-based courses. The system based on a domain model authored by the instructional designer and on individual learner models, provides to students navigation support adapted to their own knowledge and needs. WEAR also deals with modelling not only students but also the other class of its users: the instructors. By combining the information of all of the above-mentioned models, WEAR besides providing adaptive navigation to students, it supports instructors in authoring consistent and efficient courses.

6. References

[1] C. Boyle, and O. Encarnacion, "Metadoc: An adaptive hypertext reading system", *User Modeling and User-Adapted Interaction*, 4, Kluwer, The Netherlands, 1994, pp. 1-19.

[2] P. Brusilovsky, "Methods and techniques of adaptive hypermedia" *User Modeling and User-Adapted Interaction*, 6 (2-3), Kluwer, The Netherlands, 1996, pp. 87-129.

[3] P. Brusilovsky, "Course Sequencing for Static Courses? Applying ITS Techniques in Large-Scale Web-Based Education", in G. Gauthier, C. Frasson and K. VanLehn (eds.): *Intelligent Tutoring Systems, Proceedings of the 5th International Conference on Intelligent Tutoring Systems*, Lecture Notes in Computer Science, 1839, Springer, Berlin, 2000, pp. 625-634.

[4] P. Brusilovsky, and J. Eklund, "A Study of User Model Based Link Annotation in Educational Hypermedia" *Journal of Universal Computer Science*, 4 (4), Springer Science Online, 1998, pp. 429-448.

[5] P. Brusilovsky, J. Eklund, and E. Schwarz, "Web-based education for all: A tool for developing adaptive courseware", *Computer Networks and ISDN Systems*, 30 (1-7), 1998, pp. 291-300.

[6] P. De Bra, and L. Calvi, "AHA: a Generic Adaptive Hypermedia System", in P. Brusilovsky and P. De Bra (eds.): *Proceedings of 2nd Adaptive Hypertext and Hypermedia Workshop* at the 9th ACM International Hypertext Conference - Hypertext'98, Computing Science Report No. 98-12, Eindhoven University of Technology, Eindhoven, 1998, pp. 5-11.

[7] M.W. Goldberg, S. Salari, and P. Swoboda, "World Wide Web - Course Tool: An environment for building www-based courses" *Computer Networks and ISDN Systems*, 28, 1996, pp. 1219-1231.

[8] H. Hohl, H. Böcker, and R. Gunzenhäuser, "Hypadapter: An Adaptive Hypertext System for Exploratory Learning and Programming", *User Modeling and User Adapted Interaction*, 6 (2-3), Kluwer, The Netherlands, 1996, pp. 131-155.

[9] T. Murray, J. Piemonte, S. Khan, T. Shen, and C. Condit, "Evaluating the Need for Intelligence in an Adaptive Hypermedia System", in G. Gauthier, C. Frasson and K. VanLehn (eds.): *Intelligent Tutoring Systems, Proceedings of the 5th International Conference on Intelligent Tutoring Systems*, Lecture Notes in Computer Science, 1839, Springer, Berlin, 2000, pp. 373-382.

[10] T. Murray, T. Shen, J. Piemonte, C. Condit, and J. Thibedeau, "Adaptivity in the MetaLinks Hyper-Book Authoring Framework", in C. Peylo (ed): *Proceedings of the International Workshop on Adaptive and Intelligent Web-based Educational Systems* (held in Conjunction with ITS 2000), Technical Report of the Institute for Semantic Information Processing, Osnabrück, 2000, pp. 61-72.

[11] R. Nkambou, C. Frasson, and G. Gauthier, "A new approach to ITS-curriculum and course authoring: the authoring environment", *Computers & Education*, 31, Elsevier Science, 1998, pp. 105-130.

[12] M. Virvou and M. Moundridou, "A Web-Based Authoring Tool for Algebra-Related Intelligent Tutoring Systems" *Educational Technology & Society*, 3 (2), 2000, pp. 61-70.

[13] M. Virvou and M. Moundridou, "Modelling the instructor in a Web-based authoring tool for Algebra-related ITSs", in G. Gauthier, C. Frasson and K. VanLehn (eds.): *Intelligent Tutoring Systems, Proceedings of the 5th International Conference on Intelligent Tutoring Systems*, Lecture Notes in Computer Science, 1839, Springer, Berlin, 2000, pp. 635-644.

[14] WBT Systems, TopClass, http://www.wbtsystems.com

[15] H. Wu, G.J. Houben, and P. De Bra, "Authoring Support for Adaptive Hypermedia Applications", in B. Collis, and R. Oliver (eds.): *Proceedings of ED-MEDIA 99, World Conference on Educational Multimedia, Hypermedia & Telecommunications*, AACE, Charlottesville VA, 1999, pp. 364-369.