In this paper we present a collaborative system designed to develop problem solving skills in learners through problem-centric exercises. This system is part of a data collection study consisting of hundreds of students which has been incorporated into the curriculum of a signal processing course. The purpose of this study is to examine a large population of students learning about DSP as they work through concept-specific problems. By tracking student problem-solving sessions, we aim to data mine the results to model student comprehension and characterize learning preferences.

Index Terms— Intelligent Tutoring System, Web-based Learning Tools, Collaborative Filtering

1. INTRODUCTION

Successful problem solving is a process of reduction and reconstruction. Reduction involves chunking the problem into sub-problems, such that known and unknown pieces of information can be easily identified within the smaller sub-problems. Reconstruction is a process of assembling relevant pieces of information such that the solution can be reconstructed from the available pieces of information.

This problem solving task, however, will differ among learners because students approach problems with varying levels of conceptual understanding and with possibly different learning styles. Given a large collection of student solutions, it is advantageous to construct a database of user solution profiles for some canonical problems, which then can be used to assess in a comparative manner the problem solving approach of a learner who is new to the system.

This approach of identifying users with similar “tastes” (in this case, a “taste” for solving a problem) is known as collaborative filtering [1], since “taste” predictions are filtered (inferred) from a collaboration of multiple users.

In this paper, we present a system tailored for application of collaborative filtering techniques to statistically build an assessment of student knowledge by predicting a student’s conceptual understanding from a large database of user interactions with many concept-centric exercises. The objective of the system is twofold: (a) to automatically build a predictive model of preferences, via user profiles, for solving a problem, and (b) to track each student’s conceptual understanding using these predictive models.

2. PROBLEM REDUCTION

Solving a problem begins with an understanding of the salient characteristics of the problem, which in engineering problems consist of representative nomenclature, assumptions, equations, images, or visualizations. These complex knowledge constructs are known as concepts. A natural hierarchy exists in which advanced concepts are described by their relationship to simpler concepts. For example, fundamental pieces of knowledge might be as simple as a list of relevant keywords.

Suppose, for instance, that a student attempts to solve a problem in a completely unknown knowledge domain. The problem is difficult to chunk because key information within it is simply unknown. Solving this hypothetical problem involves searching for definitions of the unknown terms, and for nomenclature associated with any equations, diagrams, or plots. This problem solving process is synonymous with recursive searches, where each chunk of information is searched for and explained away, by possibly performing more subsequent searches on the given explanations.

3. PROBLEM RECONSTRUCTION

In problem-centered learning, students are tasked with the goal of solving a problem that requires they develop a framework that consists of the most relevant pieces of information for that problem. In effect, such a knowledge domain can be represented by a graph of concepts inter-linked according to association and context; this graph constitutes a concept map [2]. For example, it is common to describe the problem by using a model, and then to apply the model correctly to obtain the solution. The model can be represented by the conceptual framework of a concept map. Thus, in order to solve a problem, relevant concepts must first be judiciously chosen.
and then assembled in a proper construct, such that they lead to the correct solution.

In effect, problem solving is a process of constructing a conceptual representation of how various pieces of information connect, resulting in a mental concept map. Within a problem statement, once all unknowns have been defined, a user organizes relevant information into a conceptual framework, which consists of an agglomeration of various chunks of information whose interdependencies define both context and application. Expert solvers are able to identify salient concept(s) whose application proves to be the determining key to the solution. Understanding which concepts are significant narrows the number of interdependencies which need to be considered by the solver, which in turn reduces the number of possible approaches.

4. INTELLIGENT TUTORING SYSTEM

We have implemented a web-based Intelligent Tutoring System (ITS), Fig. 2, specifically for an introductory signal processing course. The material presented in this course follows the organization of its textbook [4]. Each chapter of the textbook explores an important set of signal processing concepts, hence the knowledge represented by ITS is structured according to the content and chronology of these chapters.

4.1. Dependency Map

Chapter concepts can be further atomized to obtain keywords, which can serve as meta-data for tagging textbook resources and related groups of problems. Hence, for each chapter, we obtain a set of tagged resources in the form of paragraphs, equations, images, and keywords. Furthermore, we use these tags to label the collection of problem groups associated with each chapter. In ITS, students have the option of searching the index of a chapter to work on a concept-specific problems. This interface is depicted in Fig. 3 (top) for “Chapter 3”.

Given a course textbook, it is natural to treat the index as a list of concepts, whereby each individual word becomes meta-data, and can be used for tagging. The organization of each chapter can be graphically represented by a dependency concept map derived by an expert from the book’s index, as shown in Fig. 1. A dependency refers to an associative link of a lower-level concept on higher level concepts, and also on the temporal dependence derived from the chronological presentation of concepts in the book. A dependency concept map differs from a concept map representation in that concept dependencies are specified by the presence and the direction of an edge connecting the two concepts, indicating that there is a causal association. Strictly speaking, in concept maps, concept links have no directionality, but are labeled with a “verb” descriptor.

4.2. Data Collection

ITS maintains two databases: one for all the questions and resources, the other for student (user) records. There are more than 1500 questions available, each tagged with meta-data either by experts, or via automatic tagging where the text of the problem is scanned for keywords that match tags obtained from the index of the textbook. In addition, a variety of auxiliary resources are catalogued in the database, e.g., figures and paragraphs from the textbook, as well as equations.

The database for students keeps track of logins, and all activity by individual users across sessions. If the objective is for students to learn all the concepts in a course, this database will collect the necessary information to see which concepts are more difficult than others.
An educational platform capable of inferring students’ conceptual understanding requires the ability to collect meaningful data. In ITS, we collect the following information:

**Question Answers** for different types of questions: multiple choice, matching, short answer, paragraph, and calculated questions. In Fig. 2 (b), student feedback for a multiple choice question is presented to the user. The system does not explicitly state the answer, but rather displays the distribution of the answers for the entire class on the given question.

**Question Chapter Number** is recorded for each question in order to provide context and an anchor for the concept map. It is possible for the system to serve the same question for multiple chapters, since in subsequent chapters, the same question might be easier to solve with an extended set of supporting concepts.

**Question Difficulty Rating** on a 5-star Likert scale is available for students to rate the difficulty of each question. Since the database of questions is dynamically changing, it will be useful to have ratings from users for all the questions. This will enable user-adaptive strategies to be developed in the future. In addition, the ranking has the potential to automatically filter out good questions from the bad ones [5].

**Question Timer** keeps track of the time spent by a user on a given problem. Time spent per question is a good indicator of the question difficulty and may be of value when determining instances of cheating or gaming the system.

**User Resource Labels** provide a meaningful measure of a student’s conceptual development as it tracks the way in which a learner organizes chunks of information into concepts. One approach is for the student to map definitions, equations, diagrams, and plots to specific concepts. For instance, given an equation, the student might be asked to label it with a corresponding concept from a short list of tags. In ITS, these tags include target, parent, and confuser concepts.

As an example, in Fig. 3 (bottom), ITS prompts the user to identify the most salient concept associated with the given problem statement. In addition, the system may prompt the user to identify a resource, e.g., an equation, from a list as the most effective in solving the given problem.
5. PROBLEM-CENTERED LEARNING

In ITS, we present a large number of problems to the user because it is important to solve many problems in order to learn the concepts. In fact, some research shows measurable gains when the same concepts must be retrieved during repeated testing [6]. However, when a problem cannot be solved it is helpful to get hints, so ITS has the capability to show the user relevant resources during problem solving sessions. Relevance is established by using the meta-data tags associated with the question, its chapter number, and various other resources from the book. For example, in Fig. 4, ITS is shown in a “hinting” mode, where conceptually meaningful equations are suggested by the system. In this mode, relevant book paragraphs, equations, concept definitions, and images can be presented to the user as hints. The ITS database can record these user searches to infer more about an individual student’s conceptual model, and can use them to adaptively update individual “tastes” for solving particular types of problems.

Students are effective learners when they take charge of their own learning by creating the types of tools that will help them succeed in solving a problem [7]. The platform should permit students to become responsible for content creation which would suit their learning styles, by focusing on answering questions for themselves and building and restructuring internal conceptual models according to their own understanding. An internal model of the domain knowledge varies among students, for instance students may elicit concepts in visual rather than list-based representations. Another view might be through a set of equations, or perhaps through MATLAB code, or with the aid of a GUI visualization.

6. DISCUSSION

In the end of semester surveys, students often cite a particular GUI as the imagery that helped them to approach a problem. Therefore, it is useful for students to encounter material from a number of different perspectives with extensive opportunities for reflection on the interrelationship of these perspectives. Ideally, educators seek to present all representations and form an equivalency among them. In other words, since it is not apparent which framework is easiest for students to conceptualize, educators tend to generalize and present them all. Students ought to have access to all, and choose the type of representation and resources that suits their learning style. This preference can be captured by the platform and incorporated into the student model. Statistical analysis of the data across users and concepts should reveal interesting trends. In the long run, this information about users will make it possible to test different adaptation strategies that should improve the system for different classes of users.

Our recent efforts have focused on making ITS widely available to instructors and institutions, so we have concentrated on adaptability and portability with the implementation of the Question and Test Interoperability (QTI) standard. QTI compliance allows for any learning object to be easily integrated into the system and for material from other repositories and collections to be ported into this XML format for content delivery. Currently, we are extending ITS to work on portable media devices, such as the Android system.

7. REFERENCES


