AC 2011-1576: CU THINKING: PROBLEM-SOLVING STRATEGIES REVEALED

Lisa Benson, Clemson University

Lisa C. Benson is an Assistant Professor in the Department of Engineering and Science Education at Clemson University, with a joint appointment in the Department of Bioengineering. Dr. Benson teaches first year engineering, undergraduate research methods, and graduate engineering education courses. Her research interests include student-centered active learning in undergraduate engineering, assessment of motivation, and how motivation affects student learning. She is also involved in projects that utilize Tablet PCs to enhance student learning. Her education includes a B.S. in Bioengineering from the University of Vermont, and M.S. and Ph.D. degrees in Bioengineering from Clemson University.

Ms. Sarah Jane Grigg
David R Bowman, Clemson University
Michelle Cook, Clemson University
Roy P Pargas, Clemson University / US Air Force Academy

Dr. Roy Pargas is an Associate Professor in the Division of Human Centered Computing within the School of Computing at Clemson University. In the current 2010-2011 school year, he is on sabbatical as the Coleman-Richardson Chair for Computer Sciences at the U.S. Air Force Academy in Colorado Springs, CO. He teaches undergraduate and graduate computer science courses. One of his research areas is in the development of software tools for teaching. Three such tools use the natural writing features of Tablet PCs to facilitate teaching in chemistry, in algebra and calculus, and in freshman engineering. He is co-Principal Investigator on three active NSF grants supporting this research efforts. Moreover, in collaboration with Iowa State University and George Mason University, he is co-Principal Investigator on an active Department of Education grant the goal of which is to develop software tools helping 7th and 8th grade math teachers monitor student progress and teaching effectiveness.

CU Thinking: Problem-Solving Strategies Revealed

Abstract

In order to analyze engineering students' problem-solving strategies, we are collecting work completed on Tablet PCs and analyzing the digital Ink using "tags" to identify events of interest using custom-designed software called MuseInk. The work collected includes problems completed in a first year engineering course at Clemson University (CU) specifically selected for their level of complexity, potential for multiple approaches or representations, and the level of structure and/or definition provided. A "Tag Universe," a database of procedural events, errors, and other items of interest, has been developed to tag relevant events within student work. The Tag Universe is organized into categories based on a theoretical framework of process activities used during problem solving: knowledge access, knowledge generation and self-management. Tags include items such as sketching the problem, identifying known and unknown values, manipulating an equation to solve for a desired variable, and checking the reasonableness of a solution. In addition, student errors are categorized (conceptual, procedural, and mechanical), and students' recognition of their errors are being analyzed based on signal detection theory. This identifies "hits" (student makes an error and self-corrects), "misses" (student makes an error and does not recognize it) and "false alarms" (student second-guesses a correct approach). MuseInk also allows the insertion of audio tags to document students' verbal commentaries about what they were thinking when specific events occurred. A user survey was implemented to identify ways to increase benefits to students using MuseInk. Tutorials and additional classroom activities using MuseInk were developed based on survey data for use in Fall 2010/Spring 2011.

To date, worked solutions and audio commentary for three problem sets were collected from total of 26 students (19 males, 7 females). One of the three problem sets has been tagged by our research team, and inter-rater reliability analysis was conducted to ensure consistent tagging. Tag data (written and verbal) is in the process of being analyzed in terms of relationships between tag categories and students' academic backgrounds and prior knowledge about engineering. We are beginning to define criteria for structuring problems to allow students from a broad array of prior educational experiences and academic preparation to develop effective and transferrable problem-solving skills.

While our methods, which use MuseInk as a research tool, are evolving, we are also considering how the software is being used as an instructional tool. A user survey was implemented to identify ways to increase benefits to students using MuseInk. Activities using MuseInk both inside and outside the classroom are being developed based on survey data, such as tutorials and peer feedback.

Introduction

Problem solving is considered to be an essential skill in engineering. For engineering students, successfully solving a problem involves managing the given information, understanding the problem context, and understanding the foundational mathematical skills needed to solve the problem. Educators must design instruction that guides students through problems while not revealing solutions, so they may learn this problem solving process. However, the varied backgrounds of these students make this task difficult. We are using novel digital Ink technology to determine how students with different academic preparation and prior knowledge progress in developing problem solving skills in a first year engineering program. Instructors use Tablet PCs to capture digital Ink from each student for use in the evaluative process. Software developed by our research team records both digital Ink as well as erasures so that the instructor may replay students' entire solution processes. In addition, tags can be inserted to identify features of interest in the student work, such as errors, successful and unsuccessful strategies, and misconceptions. These tags can contain either typed or audio comments, which facilitates think-aloud commentary on what the students are thinking while working through problems.

In order for learning to take place, students must structure available information to fit with prior knowledge to create useful understanding. Cognitive load theory provides a theoretical foundation for understanding students' learning through problem solving. The ease with which information can be processed in working memory is a primary concern of cognitive load theory[1]. Students with weak mathematical skills, or little prior knowledge of engineering or mathematical concepts, inherently have fewer constructs upon which to build new knowledge. These learners do not have the well-developed problem-solving schema and processing the multiple pieces of information required to solve a problem tends to overload their working memory capacity. They may arrive at correct answers to engineering problems by using inconsistent approaches while failing to gain a clear understanding of the physical meaning of the problem. Through this research, we are in the process of determining how students' prior knowledge affects their problem solving strategies, and comparing strategies between students with different academic preparation.

The purpose of this study, which is the preliminary phase of a larger project, is to:

- Elucidate how first year engineering students learn problem solving strategies by recording, playing back and tagging problem-solving steps for students' Inked problem solutions;
- Identify and evaluate successful and unsuccessful problem solving strategies, as well as errors and misconceptions, in terms of cognitive processes; and
- Provide instructors with a tool to identify patterns of inconsistent and ineffective problem solving strategies.

Future phases of the project involve correlating this information with students' prior knowledge of mathematics and engineering and student demographics, and developing and assessing instructional interventions to improve student problem solving strategies within the context of global engineering challenges.

Methods

During this initial phase of our project, the emphasis has been on the implementation of MuseInk® in the classroom to record, archive and tag students work [2], and the establishment of a coding scheme for student problem solving. Usability and effectiveness of MuseInk was evaluated primarily through surveys of students and faculty. Reliability and validity of the coding structure used to study student problem solving was assessed, and will support future components of the project (identification of error patterns and assessing effects of prior knowledge). This overall project plan is summarized in Figure 1.

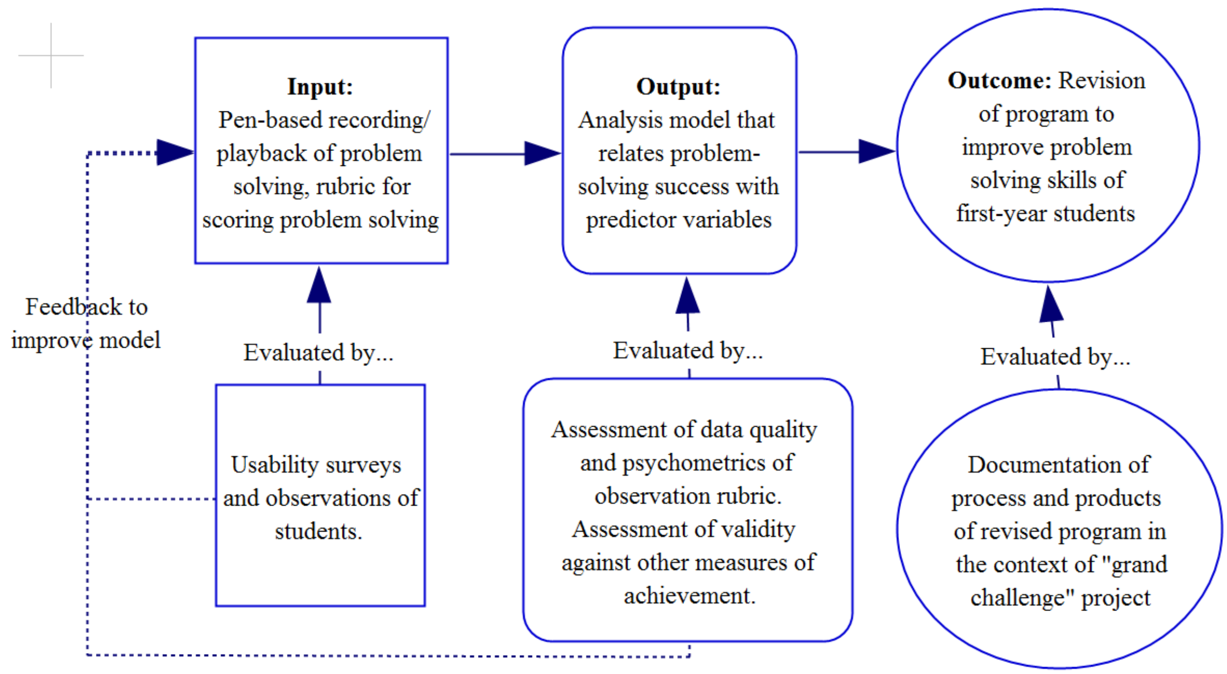

Figure 1. Summary of project plan and evaluation plan components: inputs, outputs, outcomes, and evaluation methods.

In-class Data Collection

Tablet PC software called MuseInk® has been developed by one of our research team members as a means for collecting digital Ink data. The software allows students to work problems on a Tablet PC, and stores the digital Ink in such a way it can be played back, annotated and queried. Students work through problems much as they would with pen and paper, with the added benefit of having electronic access to their work. After completing problems, students either uploaded their work to the online course interface or emailed their work to the instructor. The instructor provided feedback and grades on each problem, and archived the work for further analysis. Students were surveyed on their attitudes and behaviors toward the use of tablet computers and the MuseInk software technology each semester[3]. Based on their responses, upgrades to the software and adjustments in how the tablets were used in the classroom were implemented. For example, students reported that they did not download MuseInk to their own computers, which would prohibit them from using MuseInk to review their work, so a series of problems which were worked out and annotated in MuseInk were created by the instructor as a resource for students. In order to use the materials, students would have to download the software, which could potentially increase the numbers of students using it to replay and even annotate their own work.

The participants in this study are first year engineering students enrolled in an introductory course, "Engineering Discipline and Skills". Students enrolled in the sections of the course taught by members of the research team were invited to participate. Typical sections of the course have up to 75 students, and are taught in a "studio" setting using active cooperative learning techniques. Students are encouraged to work with their peers on in-class activities; however they complete these assignments individually.

Problems were selected for use in this study that were complex enough to allow students to discover their own way of solving the problem based on their conceptual and procedural understanding of the problem and their problem solving skills. Selected problems were well-structured [4], yet enabled the investigation of cognitive and metacognitive features of student work. A typical example is a multi-stage solar energy conversion system for which students had to calculate the efficiency of one stage given input and output values for the other stages. The problem involved organizing given information, matching that information with a schematic provided, and determining what the knowns and unknowns were. They then had to recall the equation for efficiency and solve. Students were assigned the problem in class, then were given open-ended time to solve it along with several other problems during the class period.

Data Analysis

The tablet PC software, MuseInk®, is also used for the analysis of problem solving solutions. The software allows students to work problems on a Tablet PC, and stores the digital Ink in such a way that our research team then can evaluate it. Student work (including all Ink strokes and erasures) is played back using controls similar to a DVD player. MuseInk® allows the insertion of "tags" into the digital Ink to identify significant events (errors, corrections, misconceptions, etc.). Thus procedural and conceptual problem solving knowledge can be assessed, for both research and instructional purposes. We have developed a coding structure, or "tag universe," which breaks down conceptual understanding into its two main sources of failures, true misconceptions of the problem and mismanaging the information given in the problem statement. The tag universe consists of codes, or tags, comprised of a database of errors, procedural events, concepts, and other items of interest that are relevant in problems in a freshman engineering course. These include items such as sketching the problem, identifying known and unknown values, manipulating an equation to solve for a desired variable, and checking the reasonableness of a solution. Tags are organized into categories of cognitive tasks: knowledge access, knowledge generation, and management. The coding structure is based on cognitive codes used to assess problem-solving in high school mathematics students [5]. Error tags are categorized as management errors (such as inconsistent units), conceptual errors (such as misuse of a governing equation) and "mechanics" errors (such as incorrectly manipulating an equation). MuseInk® also allows the insertion of audio tags to document students' verbal commentaries about what they were thinking when specific strokes or erasures occurred. We used MuseInk® in a first year engineering course to collect student work in class, and subsequently have students reflect on their problem-solving strategies post-hoc. All students in two sections of the course used MuseInk® to complete work in class on tablet computers, and a selected subset of students were invited to complete post-hoc audio commentaries within 24 hours of completing the problems. Students were purposefully selected to represent a diverse cross-section of gender, race and academic preparation. The Ink for this subset of students was coded by three members of the research team. Inter-rater reliability was assessed continuously throughout the project by comparing codes for all three coders and calculating the frequency and percentages of inter-rater agreement for every code used.

Results and Discussion

To date, worked solutions and audio commentary for three problem sets were collected from total of 26 students (19 males, 7 females). One of the three problem sets was tagged by three members of our research team, and inter-rater reliability analysis was conducted by an external evaluator to ensure consistent tagging. Initial overall inter-rater agreement was 55%. The team worked through inter-rater disagreement iteratively, with the main objectives being to create agreement and maintain the integrity of the codes [6]. The team discussed how to apply certain tags that were used frequently, but were inconsistently coded. In some cases codes were revised and new codes were added, and in other cases the instances in which the codes are applied were clarified. Additionally, codes that were used infrequently and/or never mutually identified were discussed in terms of the difficulty in identifying these codes, and whether or not they were useful. These discussions resulted in a revised coding structure which was used to recode the problem. The second round of coding resulted in decreased inter-coder agreement, at which point the team implemented inter-rater checking whereby a second coder reviews a first coder's work and makes editing suggestions, then the first coder reviews the edits and accepts or rejects them. After this technique was implemented (round 3), the coding structure achieved an average of 92% inter-coder reliability between the three coders for problems completed by three students. Table 1 shows the development of coding results across three iterations.

Table 1: Inter-rater agreement for rounds 1, 2 and 3, for codes identified within problems completed by three students (randomly selected from study participants).

Student #	Round 1	Round 2	Round 3
1	77%	73%	100%
2	55%	40%	96%
3	42%	25%	85%
Total	**55%**	**41%**	**92%**

The second and third problem sets are in the process of being coded by the three team members using the same coding scheme, and inter-rater reliability will be evaluated on these for three students' problems.

Conclusions and Future Work

We have developed methodology to analyze student problem-solving skills and how they develop by capturing digital Ink and post hoc audio commentaries. Through an iterative process of code structure development, we have defined a set of codes or "tags" with which these data are analyzed.

Future work includes the analysis of tag data from both written work and audio commentaries in terms of relationships between tag categories and students' academic backgrounds and prior knowledge about engineering. Through this study, students from a broad array of prior educational experiences and academic preparation can develop effective and transferrable problem-solving skills.

Acknowledgements

This work is supported by the National Science Foundation Award Number EEC-0935163, "CU Thinking". We would like to acknowledge Dr. Brandon Olszewski with the International Society for Technology in Education for conducting the inter-rater reliability analysis.

References

1. Sweller, J., Van Merrienboer, J. J. G., and Paas, F. G. W. C. (1998). Cognitive architecture and instructional design. *Educational Psychology Review* 10:251-296.

2. Bowman, D. and L. Benson. (2010) MuseInk: Seeing and Hearing a Freshman Engineering Student Think. *2010 ASEE Annual Conference Proceedings.*

3. Grigg, S., Bowman, D. and Benson, L. (2011). Technology Adoption Behaviors in a First Year Engineering Classroom. *2011 ASEE Annual Conference Proceedings*.

4. Jonassen, D.H. and Tessmer, M. (1997). An outcomes-based taxonomy for instructional systems design, evaluation, and research. *Training Research Journal* 2:11-46.

5. Wong, R.M.F., Lawson, M.J. and Keeves, J. (2002). The effects of self-explanation training on students' problem solving in high-school mathematics. *Learning and Instruction* 12: 233–262.

6. Olszewski, B., Macey and Lindsrom (2006). The practical work of coding: An ethnomethodological inquiry. *Human Studies* 29:363-380.