Work-in-Progress: The Effects of Concurrent Presentation of Engineering Concepts and FEA Applications

Prof. John Martin, Youngstown State University

John Martin is an Assistant Professor of Mechanical Engineering at Youngstown State University. John has seven years of mechanical engineering experience.

Mrs. Anna M. Martin, Kent State University

Anna Martin is a doctoral student of Educational Psychology and Instructional Technology at Kent State University and a high-school social studies teacher at Canfield High School with 9 years of experience.
Work-in-Progress: The Effects of Concurrent Presentation of Engineering Concepts and FEA Applications

John D. Martin  
Youngstown State University

Anna M. Martin  
Kent State University

Introduction

Computer-based instructional materials are a necessary component of many practical applications of engineering knowledge that is used across many engineering domains. Understanding computer-based applications will help prepare students to analyze and problem-solve engineering issues while also utilizing industry-based software. In order to most effectively present instructional material encompassing computer technologies, instructional designers must consider human cognitive architecture. Existing research regarding cognitive load theory (CLT) has examined the ways in which information should be presented so as to maximize learning while reducing strains on a student’s working memory capacity. However, a gap in the current understanding of the ideal instructional delivery methods for computer-based materials lies in the sequencing of learning content-based concepts and computing software, especially when considering the current knowledge base of the student.

Purpose Statement and Research Question

As a result of existing gaps in the research on CLT, this study will examine the effect of concurrently learning engineering software skills and engineering concepts, while applying those software skills to the engineering concepts. More specifically, we will focus on one research question: Does concurrent presentation of engineering software information affect student learning of engineering concepts for students with high-levels of existing software knowledge?

Review of Literature

This study will be approached from Mayer’s cognitive theory of multimedia learning. This theory has been built off of progress made by Paivio and his dual coding theory. Dual coding theory suggests that all information is coded into two systems (i.e., verbal and imaginal). This theory is also based on Baddeley’s working memory model. This model suggests that working memory has three components, including the executive control system, articulatory loop, and the visual-spatial sketchpad. Sweller’s work in CLT also played an influential role in the development of Mayer’s cognitive theory of multimedia learning, which provides insight into the ways in which content and presentation style can overload students’ limited working memory capacity. Cognitive overload is specific to the individual, and recent research suggests that one of the main features of this overload involves the individual’s prior knowledge base. Understanding human cognitive architecture allows for greater instructional efficacy.

Specifically, the instructional delivery of course materials should differ for novice and expert learners in order to maximize learning. Typically, novices are in need of more
scaffolding, due to their underdeveloped schema regarding domain-specific content, while experts are able to dive right into more complex tasks, due to having increased working memory capacity.\textsuperscript{1} The presentation of new information may rapidly overwhelm a novice who lacks prior knowledge, whereas experts are not as easily overwhelmed due to the existing structures in their long-term memory.\textsuperscript{3}

In fact, when experts are presented with information in the same way as novices, research suggests they may experience the “expertise reversal effect,” where the presentation of information is redundant and thus has a negative effect on learning.\textsuperscript{4} At this point, a great deal of research has assessed the negative effects of CLT and student learning, however there is a very limited amount of research that examines the simultaneous learning of computer software and engineering concepts. Prior research has identified the benefits of the sequential presentation of information to learners with low-levels of computer-based technology knowledge, while much less is known about the presentation of computer-based materials to those that are more experienced.\textsuperscript{2,4} Better understanding of the way in which instructional design impacts learners of varying levels of expertise has the potential to better guide efficient instructional planning and delivery.\textsuperscript{4}

Design/Methods

Given the limited understanding in the literature about learners with advanced technology skills, we will utilize a quantitative quasi-experimental pretest-posttest study to gain a better understanding of the effects of a concurrent presentation of finite element analysis (FEA) skills and application of engineering concepts, specifically the finite element method (FEM), on student learning. Mechanical Engineering Technology students within a required undergraduate FEA design course who have demonstrated knowledge of the software will be divided into two groups, where instructional design will differ between simultaneous and sequential presentation of information. Additionally, a subjective measure of cognitive load will be used to quantify between-group cognitive load, while a posttest measurement will assess student learning of both software skills and engineering knowledge. Participation in the sequential and concurrent instructional groups will serve as the independent variables; while the dependent variables are engineering concept knowledge, FEA skills scores, and subjective cognitive load scores.

Pre-Test: FEA Software Skill and Engineering Concepts

Students will first complete a pre-test to identify their baseline FEA software skills and knowledge of FEM. Figure 1 shows an example of a sample pre-test question where the deflection values need to be determined for an axially loaded steel plate, by means of FEM and FEA software utilizing beam elements.\textsuperscript{6}
Group 1: Simultaneous Presentation

Students in this group will have information presented simultaneously. That is, both the FEM material and the FEA software skill will be taught at the same time regarding beam analysis. While in a computer lab, this group will experience the presentation of beam elements in FEA alongside of FEM, using both to analyze an axially loaded beam solving for deflection values. Figure 2 shows an example of the simultaneous presentation of material via PowerPoint. This slide shows the concurrent presentation of FEA and FEM material, where a new FEA software skill, beam elements, is being learned and also applied to the new FEM material being learned. Following the presentation of information, students will complete the post-test, which will also include the cognitive load assessment measures.
Figure 2. Simultaneous FEA software and finite element method presentation example.\(^6\)
Reprinted from *Finite element analysis: theory and application with ANSYS*, (p. 9, 11, 12), S., Moaveni, 2015, Pearson.

Group 2: Sequential Presentation

Students in this group will have information presented sequentially. That is, students will learn the FEM content and FEA software skill independently of one another. Students will first spend multiple class periods learning only about FEM while solving for deflections of an axially loaded beam. Following this, students will then spend multiple class periods learning how to utilize beam elements in the FEA software to solve for the deflections of the same and/or similar problems. There will be a marked transition between learning of the content and software. Figure 3 shows an example of the sequential presentation of material via PowerPoint.\(^6\) These slides show samples of two separate presentations; one covering FEM and the other covering FEA software. Following the presentation of information, students will complete the post-test, which will also include the cognitive load assessment measures.
Figure 3. Sequential FEA software and FEM presentation example. Reprinted from *Finite element analysis: theory and application with ANSYS*, (p. 9-13, 16, 17), S., Moaveni, 2015, Pearson.
Post-Test: FEA Software Skill and Engineering Concepts

During the next class (i.e., following the instructional period), both groups will be given the same instructions to complete a series of post-test tasks. During each task, students will also record how difficult they perceive the task to be (i.e., cognitive load assessment). Figure 4 depicts an example of a post-test task.

![Image of a thin steel plate with deflections](image)

**Figure 4.** Post-test task example to be completed by simultaneous and sequential groups. Reprinted problem from *Finite element analysis: theory and application with ANSYS*, (p. 56) S., Moaveni, 2015, Pearson.

Post-Test: Measuring Cognitive Load

Subjective measures of cognitive load will also be used to measure students’ perceived difficulty when performing the post-test tasks. Students will rate their perceived mental effort based on a 7-point Likert scale, ranging from 1 - *extremely easy* to 7 - *extremely difficult*, after
interpreting the instructions for each post-test task. This approach has been used in prior research, and has been shown to accurately gauge the amount of mental effort exerted by participants.7,8

Implications

Student understanding of both engineering concepts and the associated engineering software encourages their ability to apply both aspects of knowledge to the types of problems encountered in industry. In order to meet the demands of individual learners, a better understanding of the way in which the sequencing of instruction impacts student learning is necessary. This study seeks to shed light on the ways in which instructional delivery impacts student learning of engineering software skills and concepts.

References