Advances in Engineering Education

2025: VOLUME 13 ISSUE 2

A Usability Study for an Online Engineering Mechanics Exercise System with Automated Feedback

ARINJOY BASAK

NICOLE PITTERSON

CLIFFORD A. SHAFFER

SNEHA PATEL DAVISON

DAVID A. DILLARD

JAKE GROHS
Virginia Tech

ABSTRACT

Motivated by the benefits of repeated deliberate practice, we created an interactive exercise system for use in an undergraduate engineering mechanics class that focuses on practicing learned fundamental concepts. These exercises take the form of traditional word problems commonly found in mechanics courses, involving things like selecting and solving the correct equations. They are automatically graded, and the system provides targeted feedback that identifies errors in student solutions. To evaluate the success of the design of our interface and the feedback engine, we conducted usability studies with students in two offerings of an engineering mechanics course in Fall 2020 and Spring 2021. We also surveyed instructors teaching similar course content at other universities to evaluate our system. Our instructor survey collected input on what features they feel would be pedagogically useful based on their experiences teaching similar courses. Our studies showed instructors and students liked the core interface design, and instructors found the system easy to learn. However, students and instructors requested improved interface intuitiveness and usability in general, with specific requests for flexibility in equation inputs and workspace management. Students and instructors favored the feedback system in general, with students expressing mixed reviews, and instructors expressing concerns that too much feedback could interfere with learning. We use these findings to provide an outline for future development plans for this system.

INTRODUCTION AND BACKGROUND

Engineering expertise is often defined as the ability to solve complex and ill-structured problems. Engineering courses at all academic levels typically use embedded problem-solving activities for disciplines such as mechanical, biomedical, civil, aerospace and ocean engineering, and mechanics concepts are a fundamental part of the curriculum. However, years of research have demonstrated that students continue to experience difficulties understanding mechanics at the conceptual level (Streveler et al. 2008; Watson et al. 2016; Nelson et al. 2017). Conceptual change researchers have attributed the ensuing difficulties associated with learning basic mechanics to the following factors: 1) insufficient mathematical knowledge, 2) overall abstractness of the content, 3) students' preconceptions of the content and 4) the degree of logical precision required in problem solving (Johnstone 1991; Georghiades 2000; Pitterson 2015; Lemke 1997). To combat these factors, researchers have recommended the use of multiple representations of the concepts as well as opportunities for repeated practice (Ainsworth 2008). Additionally, the use of technology-enabled tools has been reported to significantly reduce the cognitive gap associated with learning fundamental concepts such as mechanics (Dori and Belcher 2005; Ertmer and Ottenbreit-Leftwich 2013).

Despite the prevalence of research findings and recommendations associated with the implementation of evidence-based instructional approaches in teaching, some studies continue to highlight that there is a mismatch between what is taught, what is learned, and what is assessed (Wiliam and Thompson 2017; Reinholz 2016). Further, investigations with validated physics and mechanics concept inventories have identified that students' conceptual understanding is in stark contrast to their achievement in courses (Sands et al. 2018; Yang et al. 2020; Montfort et al. 2015; Steif and Dollár 2005). Qualitative studies used to investigate this phenomenon have shown that students at varied academic levels often demonstrate expected proficiency in problem-solving, but that conceptual understanding is inadequate (Yang et al. 2020). That is, students who progress in their studies become better at calculating solutions to well-structured problems, but some remain deficient in the conceptual principles required to reason through complex or novel problems.

The foundations of teaching engineering mechanics have been extensively researched. Researchers have noted issues with traditional means of teaching engineering mechanics, namely students' difficulties in connecting fundamental concepts to solving problems (Steif and Dollár 2005), and recommendations for organizing core topics in the syllabus. There has also been work in documenting foundational concepts and skills learned in engineering mechanics, particularly in statics (Steif 2004). Extended work in this area such as the development of relevant concept inventories (Steif and Dantzler 2005) has improved instructional development in the field (Dollár and Steif 2008; Steif and Dollár 2009). Researchers have also explored how student problem solving skills connect to future success (Kirn and Benson 2018), and in statics have shown differences among students' learning approaches (Litzinger et al. 2010).

Our NSF-funded project is rooted in the belief that problem solving is foundational to engineering education, but that growing class sizes and demands on teaching time, as well as students' perceived lack of sufficient prior knowledge, have deemphasized aspects of problem solving that research on learning and use of evidence-based pedagogical practices have demonstrated are crucial for knowledge transfer. Educational researchers argue that technology-rich learning environments can be used to overcome these challenges and thus foster conceptual understanding (Lund, Hauge, and Hauge 2011; Halverson and Graham 2019; Azevedo and Gašević 2019). To systematically investigate how a technology-rich problem-solving interface can enhance the teaching, learning, and assessment of complex engineering knowledge, researchers must initially develop prerequisite understandings of both the processes by which students are actively constructing knowledge in a specific domain (Koohang et al. 2016), and the critical factors that either facilitate or undermine such active construction (Ioannou, Demetriou, and Mama 2014; Steif and Dollár 2009).

Design and evaluation of educational technologies is an active field of research that has led to insights on how successful software for such applications should be designed. Drijvers (Drijvers 2015) states several crucial factors such as design, role of the teacher and educational context, and particularly how crucially the pedagogical functionality in which a tool is incorporated must match the tool's characteristics. From a design perspective, Oppermann (Oppermann 2002) outlined well-accepted standards for user-interface design for learning systems. The use and acceptance of such systems by the two important stakeholders — instructors (Beggs 2000) and students (Pierce and Stacey 2001; Popovici and Mironov 2015) — have also been studied at length, to identify key factors that would make new software attractive to users. There also exist theories that explain what makes software attractive for widespread adoption (Rogers 1995), such as the Unified Theory of Acceptance and Use of Technology (UTAUT) model (Venkatesh et al. 2003).

Feedback plays a critical role in learning as it is the process by which students are given information about their understanding and performance on assessment measures. However, providing good quality feedback is difficult in large classes. We note the key issues that frustrate students are: i) the delay in receiving feedback and ii) the inconsistent quantity and helpfulness of the comments (Hounsell et al. 2005). Additionally, the nature of the feedback comments provided is essential for improving students' understanding of the material, be it in terms of improving self-regulated learning skills (Van Den Boom et al. 2004; Chen, Whittinghill, and Kadlowec 2010), or introducing effective pedagogical strategies based on context (Chi et al. 2010). Additionally, brief and compact feedback has shown marked learning improvements (Di Eugenio et al. 2008) over automated systems that generate overly repetitive feedback.

SYSTEM OVERVIEW

To address the issues raised above, we have designed and implemented an interactive problem-solving tool aimed at improving students' conceptual understanding of fundamental mechanics concepts through deliberate, repeated practice and targeted feedback. The system is built using support libraries that are part of the OpenDSA eTextbook system (Shaffer, Naps, and Fouh 2011; Shaffer et al. 2011). Additional open-source libraries for mathematical solvers and physical quantity manipulation were used to implement technical aspects of the feedback system. Exercises written in HTML with a corresponding solution file can be served as standalone web pages or through learning management systems like Canvas. We use click-and-drop as the main interaction idiom. To aid new users, we provide text snippet prompts and detailed in-system help/tutorial material for every element. This information is accessible by clicking on the appropriate question-mark icon for that element (Item 7, Figure 1).

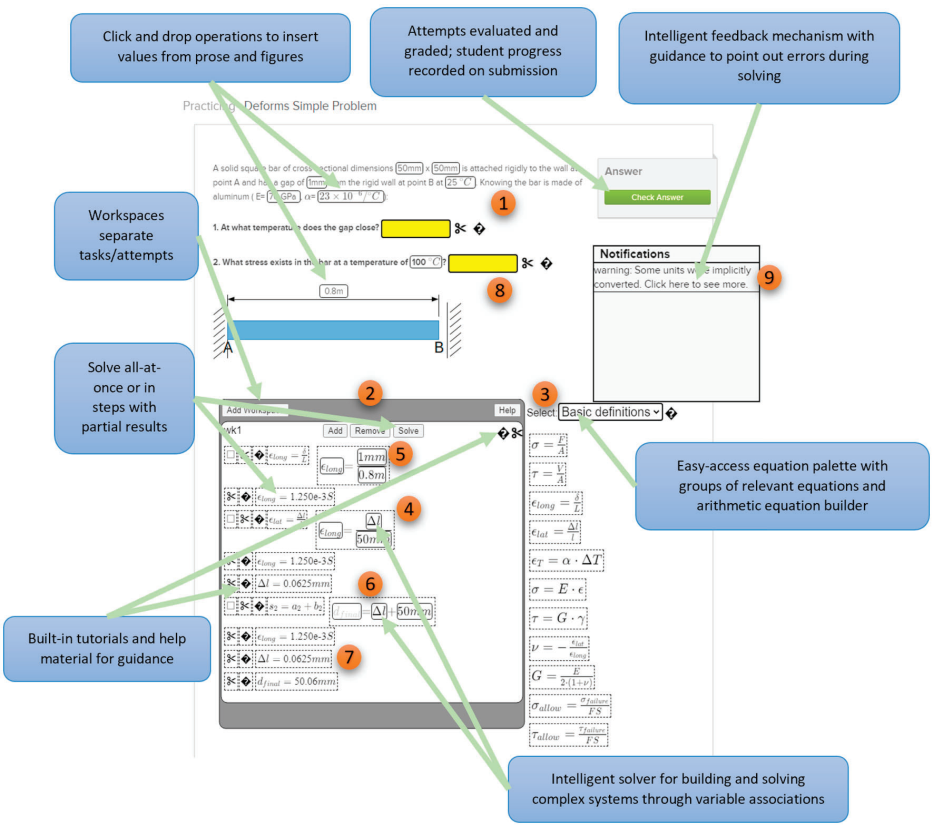


Figure 1. An outline of the problem-solving interface with the different features offered.

We present a sample exercise in Figure 1 (see also: https://opendsax.cs.vt.edu/ODSA/Books/DeformsPublicDemo) to demonstrate the key elements of the system interface. The problem prose (Item 1 in Figure 1) contains the text and diagrams related to the problem with clickable physical quantities embedded in the prose (white boxes with quantities, see Item 1) that users can click to add to equations (Item 4) in the workspace (Item 2) to solve problems, as well as question sub-parts with yellow submission boxes to receive answers. Users solve problems inside the workspaces (Item 2) by assembling systems of equations to be solved. Workspaces help keep logical separation of equations for subparts or alternative solution attempts and can be resized automatically to accommodate equations and computed answers (value boxes, see bottom of workspace in Figure 1). We leverage the benefits of palette-based entry to reduce algebraic tedium (Item 3), with equations grouped by topics in the course, along with equations for basic algebraic relations such as sum, product and ratio, and a "Favorites" list for equations recently used in the session.

Equations (Item 4) are the main workhorse for our system. Equations are added to a workspace by clicking on an equation in the palette and then clicking on the "Add" button as shown in Figure 2. Checkboxes are used to select equations to be solved as a set (click on "Solve"), or deleted (click on "Remove"), allowing to solve or remove subsets of equations. Each equation has a section with variable boxes, where users can enter predefined quantities from the prose or custom quantities, and convert their units on prompt (Item 5). We create equations in one unknown by populating all but one box in a single equation. Systems of equations in more than one unknown can be created using variable associations (Item 6 in Figure 2). To do so, we click on an empty box to start an association from its context menu, and then click on a second box. More variables can be added and customized this way (Figure 2).

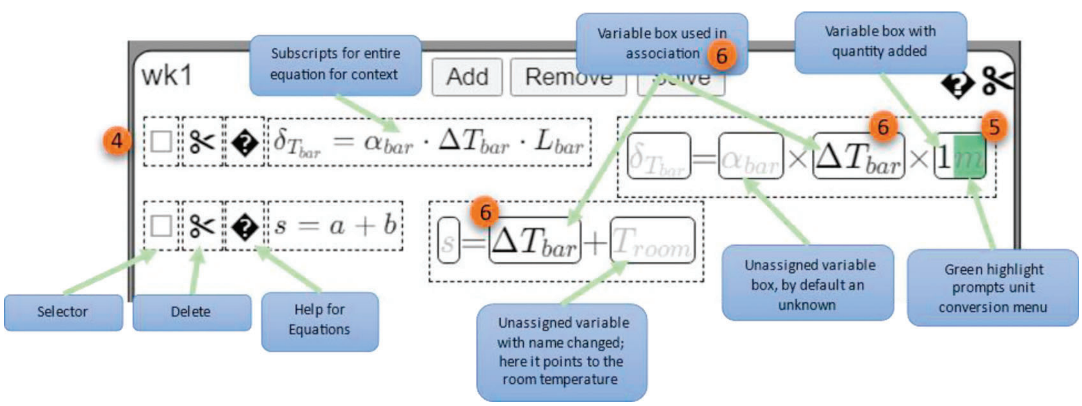


Figure 2. Features of equation objects.

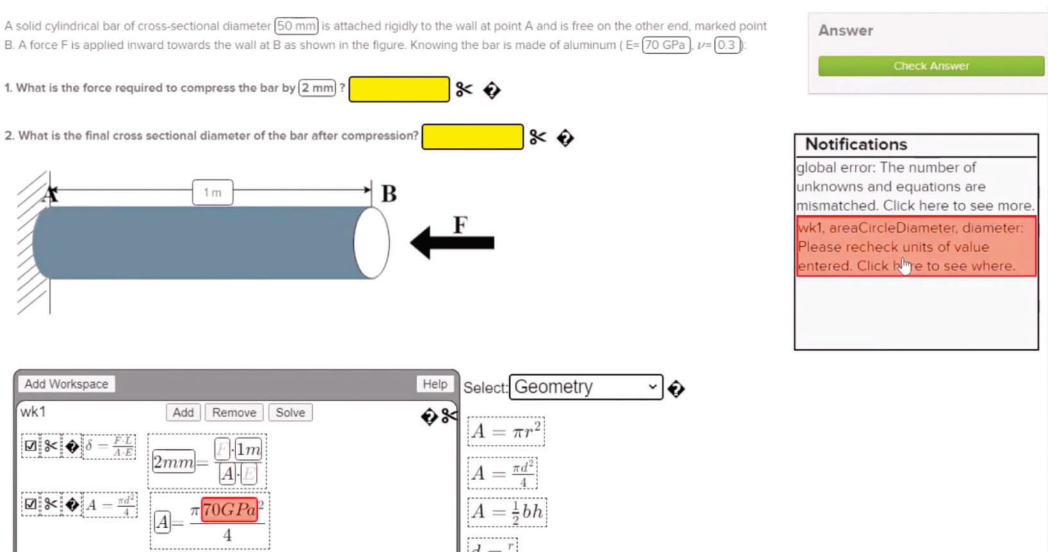


Figure 3. Example of the guidance system in action.

Our solver engine uses Nerdamer to solve equations, and math.js for working with physical quantities. Equations are preprocessed using these two libraries for use by the solver. The results of preprocessing are used by the guidance system (Item 8 in Figure 1), as outlined by the Notifications panel shown in Figure 3 to provide targeted, brief feedback. Research shows briefer feedback to be more useful than long feedback (Anderson et al. 1995). Students are directed to address blatantly wrong issues that lead to erroneous computations. The preprocessor checks for inconsistencies in units on both sides of equations, placement of improper quantities in boxes in equations, and the number of equations and unknowns. Implicit unit conversions to match orders of magnitude and inferences of units of results are also done. Errors caught in either of these two phases are reported in the Notifications panel as interactive message text. Students can then click on interactive parts of the message, and the user interface will highlight locations in the workspace where the errors occurred (which could be variable boxes corresponding to a variable association, a single variable box with a quantity, an equation itself, or a general error message.

The errors reported by the system are syntactic in nature. The types of errors that the system can detect and report include placing a quantity with unexpected units in a box in an equation, creating an equation with incompatible units on either side of the equation, creating a system with different number of unknowns and equations, and creating a variable association between boxes in separate equations that have mismatched units. Feedback is generated at the time of solving a system of equations, after the student selects a set of equations to solve, clicks on the "Solve" button, and

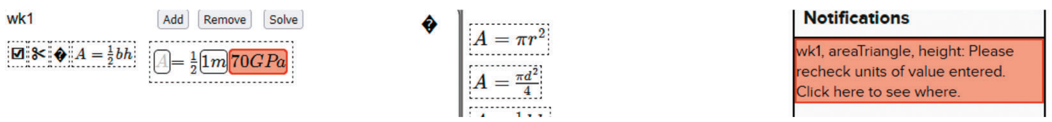


Figure 4. An example of an error message given for populating a box in an equation with the wrong unit.

the system completes the processing. In Figures 4-7, we illustrate some typical errors that students make, and the corresponding error messages that they encounter.

- A student may enter a quantity in a box in an equation template that does not match the units for the quantity that box expects. We illustrate this scenario in Figure 4. The student added the equation to calculate the area *A* of a triangle, which expects a base *b* of length units and a height *h* of length units. The student correctly adds a length quantity of *1 m* from the prose for the problem into the box for base, but wrongly adds a pressure quantity of *70 GPa* in the box for height. Consequently, an interactive error message is generated. The student can click on this to show which box has the unexpected quantity.
- A student might set up an equation in a way that has quantities with incompatible units on either or both sides of the equation. This particularly occurs when students use generic arithmetic equations, as there are many ways for students to make a mistake. We illustrate one such scenario in Figure 5, where a student attempts to add a *50* mm length quantity to a *70 GPa*

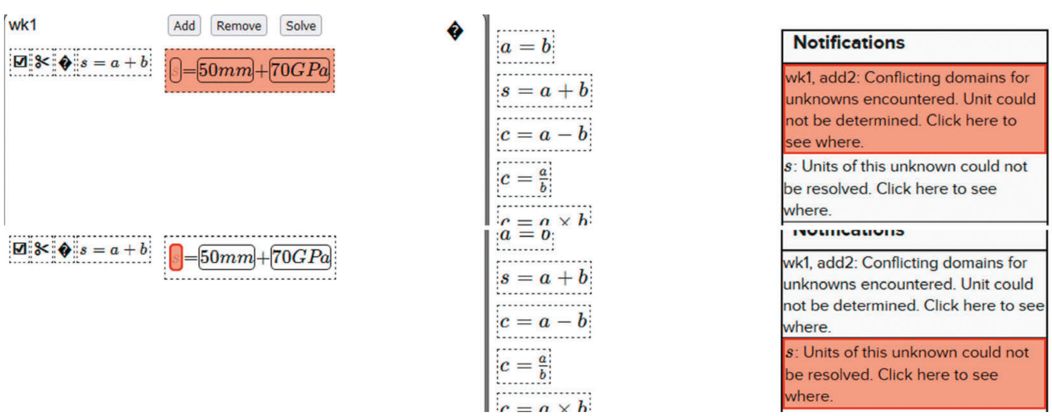


Figure 5. An example of an error message when quantities with incompatible units are used to solve an equation.

pressure quantity. Consequently, attempting to solve this equation to find sum *s* results in the error message shown. Clicking this highlights the equation where the error occurred. In this case, it makes more sense to show the equation with the error since it is not possible for the system to decide explicitly if the sum s is intended to be a length quantity (in which case, the box with the pressure quantity is wrong) or a pressure quantity (which would make the length quantity wrong). Hence, the fact that the units for s could not be determined is also reported as an error, indicating that equations that computed s are the sources for the error. Students generally solve larger systems of equations that consist of more than one equation. For such systems, even highlighting individual equations related to a specific error as opposed to more fine-grained error detection can be helpful in corrections.

- A student might associate the variables in a system of equations incorrectly, leading to a mismatch in the number of unknowns and equations, creating an inconsistent system of equations that cannot be solved. Figure 6 shows a system of two equations in two variables. We intend to use the calculated area *A* from the second equation in the first equation, so we associated the boxes marked *A* in both equations. However, there are two empty boxes in the first equation, which are treated as unknowns. This creates a system of two equations and three unknowns, which is inconsistent. Attempting to solve this system shows the message in the Notifications panel. Clicking this notification will then show the complete message as shown in the figure. This can happen if the student forgets to populate a box in one of the equations with a parameter, or associate boxes from different equations.
- A student might create a consistent multi-variable system of equations to solve but accidentally associate boxes from different equations that have incompatible units. In such a case, since the variable association represents an unknown, we cannot determine the unit of the final quantity with certainty, leading to an error. From the equation template in Figure 7,

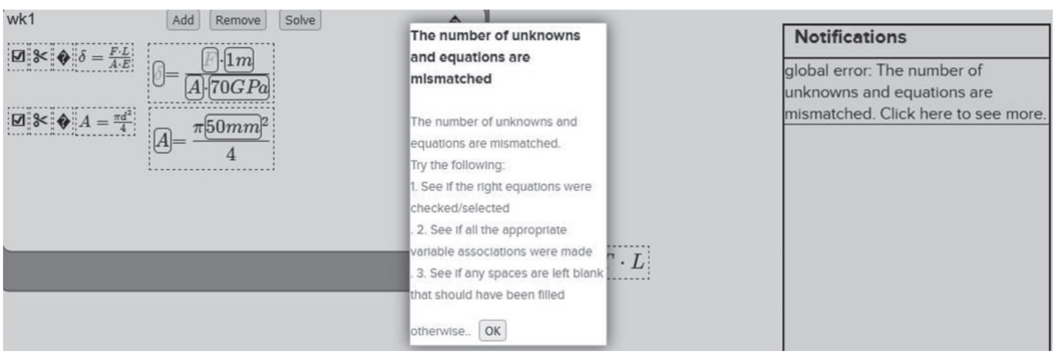


Figure 6. An example of an error message where the system of equations is inconsistent.

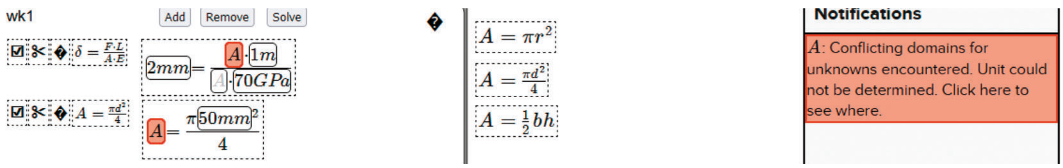


Figure 7. An example of an error message for an incompatible variable association.

we know that the student was supposed to associate the box for *A* (representing area) in the first equation to the box for the calculated area, *A,* in the second equation. Instead, the student connected the box for *A* in the second equation (which has area units) with *F* in the first equation (which has force units), leading to an incompatible variable association. This led to the system being unable to uniquely determine a unit for the unknown associated with the variable association. This is treated as an error, and a corresponding error message is shown in the Notifications panel. This message identifies the variable association and the error that occurred, as shown in the figure. Clicking this error message displays the boxes for the incompatible variable association.

In a typical workflow, a student would read the problem prose, add the appropriate equations to a workspace from the palette, populate parameters of the equation with quantities from the problem statement, and make associations as required. A completed system of equations can be selected and solved which might or might not generate errors. A student would address these errors, and then repeat the process until no more errors are generated and a candidate solution is computed and added to the solution box for a question part. Once the candidate solution is added, the student can click on "Check Answer", and the system will indicate whether the submitted answer is correct or not.

STUDY CONTEXT

ESM-2204, Mechanics of Deformable Bodies, is a sophomore-level course typically offered every semester at our university. It is organized as either two 75-minute lectures or three 50-minute lectures each week. Section capacities typically range from 50 to 100 students. The course introduces the following topics to primarily second-year students: concepts of stress, strain, and deformation; factor of safety; stress-strain relationships and material properties; stress concentrations; area moments of inertia; axially loaded members, torsionally loaded members, and bending of beams; shear and moment diagrams; stresses due to combined loading; thin-walled pressure vessels; transformation of stress including Mohr's circle; and beam deflections and buckling stability.

Data Collection and Analysis

To gather feedback from stakeholders about the system we created a survey instrument for student feedback and another for faculty feedback. For the study involving the students, we deployed four exercise problems in the Fall 2020 semester and six exercise problems in the Spring 2021 semester as extra credit assignments where students could receive credit by solving problems using our system and optionally complete a usability survey based on their experience. Non-participants could alternatively submit a solution to the same problems solved on paper as part of a practice problem set. The faculty survey was offered to instructors with experience in teaching the equivalent of ESM-2204 in other universities. Both student and faculty participants were asked to solve or attempt to solve multiple problems on the software before providing their feedback.

The student's usability survey had twenty-six questions including eighteen 5-point questions, two either-or questions, and six qualitative questions. The Likert-scale questions (Figure 8) asked about the degree of positive or negative reaction to specific interface design items. The qualitative feedback questions (Tables 1-12) were used to elaborate on their responses and collect suggestions for improvement. The student survey instrument was administered as part of user studies conducted in Fall 2020 and Spring 2021 sections of the ESM-2204 class. 162 students worked on the system as recorded by interaction logs. This was further filtered for significant engagement. This was defined as students who solved at least one problem successfully, or registered at least 200 events in the first study, or 300 events in the second study. This gave us a total of 96 students, of which n=80 students also completed the survey.

For the instructor survey, our questions (Tables 1-12) were aimed at instructors with prior practical experience in teaching the course. Our goal was to evaluate instructors' perceptions of the use case and pros and cons for the system, so that we could gauge the need for further system development. Instructors that we knew were initially approached, they were also encouraged to share this survey with other instructors. We also sent email containing links to the survey to the ASEE Engineering Mechanics listserv and to the Engineering Mechanics faculty listserv at our university. In total, we received detailed responses from 15 instructors.

The responses to the Likert-scale and either-or questions were evaluated quantitatively by observing the percentage of each response to obtain a broad view of student perception on the interface design, tutorial support, and targeted feedback. The qualitative question responses were analyzed for multiple, and sometimes overlapping, themes coded in detail. Possible broader themes were coded alongside more specific themes. Common codes reported in student and instructor responses were grouped together into larger themes, and the same process repeated. At each stage, combinations of codes were used to identify overarching themes that appeared in multiple responses. This process was repeated twice before we found thematic codes that converged to common themes. We then compared quantitative and qualitative responses from questions in different categories to obtain

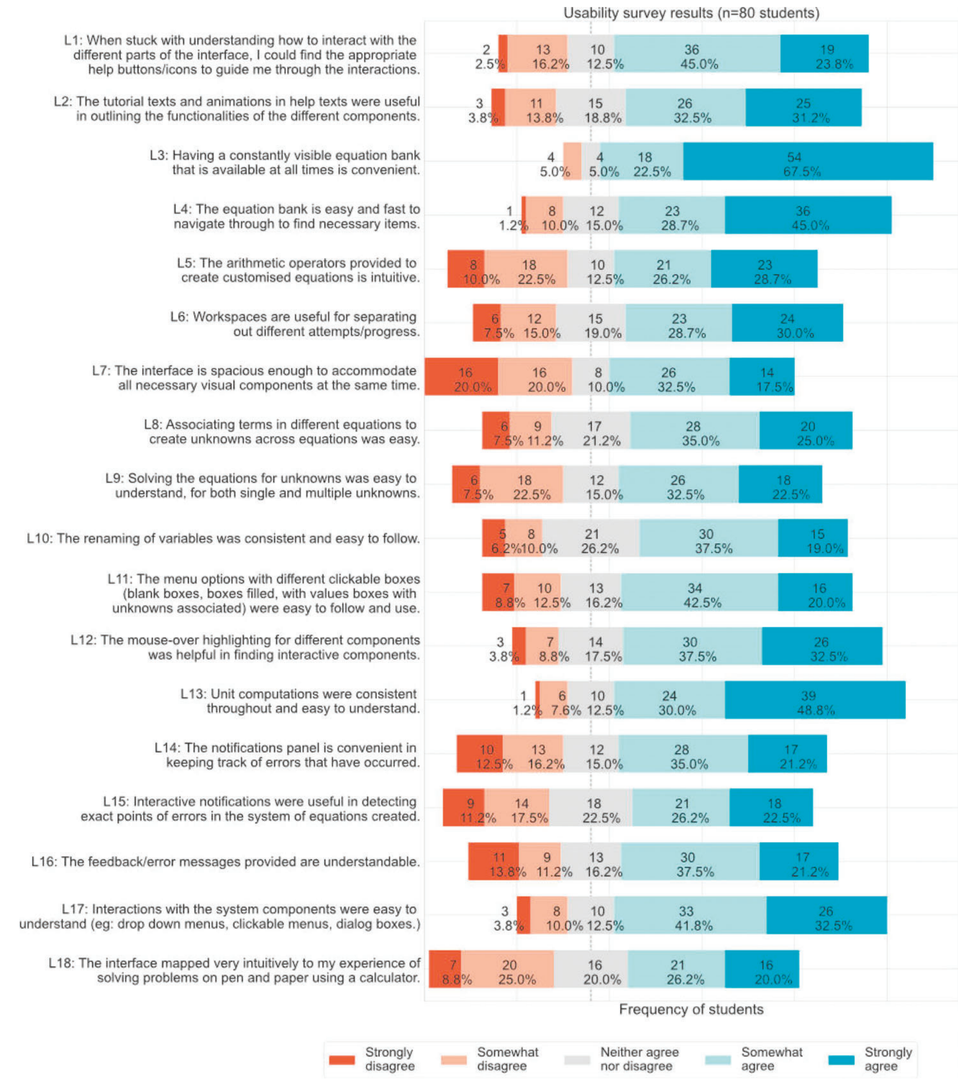


Figure 8. Student responses to Likert-scale Questions. Segment lengths reflect the distribution of sentiment among student responses.

first overall opinions for each aspect of the system. We then explored detailed themes regarding what the students and instructors liked and disliked.

RESULTS

We divide our discussion of the students' and instructors' responses into two categories based on their opinions on: i) the usability of the overall system interface for working through exercise problems, and ii) the benefits of the feedback mechanism. We briefly summarize the opinions of students and instructors and illustrate similarities and differences in opinions between students and instructors on issues related to the system and its observed usefulness. The student survey mainly focused on usability, and the instructor survey mainly focused on the impact of the system as an educational tool, so the distribution of responses in each of the categories are not necessarily related.

Opinions on Interface Design and Usability

The students were asked 15 five-point scale Likert-scale questions (L1-L13, L17-L18, Figure 8), 2 either-or questions, and 5 qualitative questions (Q1-Q3, Q5-Q6) about their experience with using the system interface to solve the exercise problems. The either-or questions and their responses are as follows:

- S1: The click and drop interactions were enough to perform the necessary tasks efficiently. Yes/No (Yes: 58/80, 72.50%, Students who said "No": 27.50%)
- S2: Would you rather prefer (Option 1) All functionalities of the system (all menu options, equation bank, etc.) available in the open, even though it might take up interface space and reduce work area, OR (Option 2) Have functionalities efficiently grouped in menus, which would increase workspace area, but would need more than one click to perform a specific function? (Option 1: 29/80, 36.25%, Option 2: 51/80, 63.75%)

The questions focused on how the design choices of the user interface and the functionalities provided (user prompts, equation solver) assisted the students in solving the problems, and how this experience compared to solving exercise problems using pen and paper. The instructors were also asked similar qualitative questions (T1, T4-T6) on the usability of the system and their opinions on using this system in their courses.

Students

Students were provided several resources such as a full online tutorial webpage demonstrating features of the interface, a walkthrough video tutorial of a single two-part exercise problem, and short text in the interface that could be accessed by clicking on the question mark icon positioned near most user interface elements. When asked to rate these features overall, student responses were mostly positive (Figure 4, L1: 23.8% Strongly Agree, 45% Somewhat Agree, L2: 31.2% Strongly Agree, 32.5% Somewhat Agree). However, when asked to elaborate on additional guidance material they would like to see (Table 1), we received mixed responses. Positive responses mentioned that they found the interface intuitive and easy to learn and to navigate, with the in-interface help being enough. Negative responses reported that the help materials were insufficient, or that information about basic workflows - such as equation setup and entering quantities - was missing. There were

Table 1. Q1. Please include any additional guidance material that you would like to see to make it easier to learn or find one's way around the system. (43/80 students responded in total, but not all responses indicated a theme.).

Themes recorded in student responses to Q1	No. of responses reporting this theme/total responses received
Explicit positive	9/43
Provided material helped in learning to use the system	6/43
Explicit negative	5/43
Help resources were lacking/insufficient	8/43
Reported usability issues with system	10/43

suggestions to include alternative modes of tutorial material such as PDFs. These responses indicate a need to improve the quality of in-system and out-of-system help material, such as language, content, and presentation of the help texts.

When asked to detail bugs that the students encountered in using the system (Table 2), three major themes were observed: bugs with manipulating equations and using generic algebraic equations (4 students); bugs with the equation solver leading to incorrect unit inference and computation, reporting inconsistent systems of equations, and issues with the numerical solver library (5 students); and arrangement of interface elements that affected readability, especially regarding the display of equations with the creation of multiple workspaces (4 students). Most of these bugs have since been fixed.

The students generally liked the design decisions overall (L3-L13, Strongly Agree and Somewhat Agree responses combined were > 50% for all questions). When asked to elaborate, they listed specific aspects of the design features that they disliked or suggested improvements. They found

Table 2. Q2. Please include any specific bugs encountered that hindered your progress while using the system. (63/80 students responded in total).

Themes recorded in student responses to Q2	No. of responses reporting this theme/ total responses received
No bugs encountered	41/63
Bugs reported	22/63
Issues with specific operations	10/22
Actual bugs	6/10
Suggestions for improving functionality	4/10
Additional bugs (menus blocking elements, improving interactions, etc.)	12/22
Workspace expansion affecting readability	4/12

Table 3. Q3. Please include any other comments about the design of the interface and if there are any new features or changes that would be of value. (50/80 students responded in total).

Themes recorded in student responses to Q3	No. of responses reporting this theme/ total responses received
Improvements to existing interface design/new design suggestions	20/50
Reduce scrolling in interface to access elements	7/20
Changes to workspaces (larger workspace, minimizing workspace, equation rearrangement)	8/20
Creating custom equations/manipulating generic equations	15/50
Improving interface usability/intuitiveness	16/50
Placement/arrangement of interface elements	4/16

the workspaces are useful (L6: 30% Strongly Agree, 28.7% Somewhat Agree; Figure 8), but did not indicate that the workspace interface was spacious enough (L7: 17.5% Strongly Agree, 32.5% Somewhat Agree; Figure 8). They suggested the layout needs to be larger or flexible enough to accommodate equations and solutions properly (Q3, Table 3). Some suggestions to improve this included a horizontal workspace layout and changing menu interactions to ensure elements did not overlap with each other. Students were asked if they would rather have functionalities hidden behind context menus that would require more clicks but increase workspace area (S1: 51/80 agreed) vs. directly accessible with dedicated buttons that might take up more workspace area and require memorizing (S2: 29/80 agreed). This suggests what improvements students prioritized.

Regarding the features for working with equations, students liked the persistent equation bank and its aid in quickly finding required equations to construct systems of equations (L3: 67.5% Strongly Agree; L4: 45% Strongly Agree, 28.7% Somewhat Agree; Figure 8), but students indicated that the arithmetic template equations provided for customized equations were not sufficiently intuitive (L5: 28.7% Strongly Agree, 26.2% Somewhat Agree; Figure 8). The operations to associate unknowns in different equations (L8: 25% Strongly Agree, 35% Somewhat Agree; Figure 8) and the process for solving systems of equations for single and multiple unknowns (L9: 22.5% Strongly Agree, 32.5% Somewhat Agree; Figure 8) had fewer negative responses. This shows that while the design choice of using equation templates from a palette for well-known predefined equations (from their course material) were perceived as useful, students wanted a faster way to create basic, non-predefined relationships between quantities and unknowns.

Students responded positively to the intuitiveness of the interface interactions (L17: 32.5% Strongly Agree, 41.8% Somewhat Agree; Figure 8). However, while students indicated that the click

Table 4. Q5. Please include other interactions that you would like to see. (Students who answered "No" to the question "Were the click and drop interactions enough to perform the necessary tasks efficiently?" responded to this question). (19/80 students responded in total).

Themes recorded in student responses to Q5	No. of responses reporting this theme/total responses received
Typing in equations and quantities	7/19
Drag and drop	2/19

and drop interactions were sufficient to complete the tasks (Yes=58, No=22), there were several suggestions for other interactions as well. In Q5 particularly (Table 4), students suggested drag-and-drop interactions, and improved ability to create customized equations by either typing them in manually or using other input methods. While the equation template was helpful, populating the equation using click-and-drop interactions and selecting individual equations to create a set of equations to solve was reported to be tedious, contradicting our original expectation that these would reduce the tedium of setup and solving systems of equations. Hence, this is an important direction of improvement for us.

The few students who responded positively to Q6 (Table 5) reported their work being faster and cleaner than the pen and paper experience, similar to what instructors reported (T1, Table 6). However, many students gave suggestions for a more intuitive UI that would allow for faster data entry. These include drag-and-drop interactions, easier equation entry and manipulation for custom

Table 5. Q6. Please include any special features that would help make the system faster and more efficient to use. (59/80 students responded in total).

Themes recorded in student responses to Q6	No. of responses reporting this theme/total responses received
Satisfaction with the system	6/59
Improvements to user interface	14/59
Request for drag and drop	5/59
Improving equation setup functionality	4/59
Improved support for working with units	4/59
Improving visuals (rearranging equations, minimizing/enlarging workspace, etc.)	9/59
Improved functionality to manipulate equations/create custom equations	10/59
Preferred to type in values and equations	8/59
Easier means to enter values	7/59
Suggestions for new features	15/59

Table 6. T1. What are your first impressions of the system? Positive attributes? Possible concerns? General thoughts? (14 responses received in total)

Themes recorded in instructor responses to T1	No. of responses reporting this theme/total responses received
Positive comments	7/14
Liked the equation banks	4/14
Might interfere with learning	4/14
Handwritten solutions have benefits over this	2/14

equations, and improved workspace layout, as outlined in Table 5. Suggestions for new features (15/59) included a search bar for equations, support for free-body diagrams, color coding of elements, support for a scientific calculator, an undo button for reversing actions, and the ability for the system to recognize typed units and variable names to be included in the system of equations.

We received mixed reviews from the students regarding the system interface. When asked to rate individual design choices for the system interface design, the responses were mostly positive (L18: 37/80 positive, 27/80 negative; Figure 8). Students expressed the need for improvement in individual design features that we plan to address in the future.

Instructors

The instructors' first impressions of the system were quite positive (T1, Table 6). They reported that the system was easy to learn (with instructors taking 1-2 problems or in some cases 20-30 minutes

Table 7. T4. In the implementation you worked with, we focused on basic stress and strain relations, axial loading and torsion problems, including statically indeterminate problems as part of a Mechanics of Materials course. Would you recommend we continue developing this platform for mechanics of materials courses? Would there be other introductory engineering courses which might also benefit from such a system? (6 responses received in total)

Themes recorded in instructor responses to T4	No. of responses reporting this theme/total responses received
Positive support for continued development	4/6
Request for support for statics problems	6/6
Support for diagrams	5/6
Support for vectors	1/6
Co-analysis of diagrams and equations for consistency and correctness	1/6

Table 8. T5. Thinking about intuitive usability of the system, what aspects worked well or didn't work well for you? (12 responses received in total)

Themes recorded in instructor responses to T5	No. of responses reporting this theme/total responses received
Positive responses (specific features liked, etc.)	4/12
Intuitive system	2/4
Improving the tutorial	2/12
Improve intuitiveness/usability	8/12
Improving specific functionalities (value input, solving equations, etc.)	4/8

of active engagement to become familiar with the system), and one comment noted the similarity to solving the problems on pen-and-paper: "It's closer to being real problem solving than anything I've seen". Equation banks were received positively, but there were some negatives about pedagogical benefits discussed later. Among aspects that need improvement (T5, Table 8), instructors reported the tediousness of using the interface for solving more complex exercise problems, suggesting that interactions such as variable association setup be made more intuitive (also requested by students in Q3, Table 3). Additional suggestions included auto-solving equations when recognized to be consistent, improving the tutorials, and better error messages (also reported by students, Q4, Table 10 and Q6, Table 5). Overall, improvements indicated a need for greater intuitiveness in interface functionality.

Opinions on Feedback System as an Educational Tool

We asked the students for their opinions on usability of the feedback system (Q4, Table 10). We also asked the instructors about how they imagine students would use the feedback system (T2, Table 11), how they themselves would deploy this system in their courses (T3, Table 12), and the courses in which they would like to see this tool be used (T4, Table 7). We captured some of their opinions on this topic when we asked about their first impressions of the system as well (T1, Table 6).

Table 9. T6. What further changes/improvements would be needed in order for you to feel comfortable using this in some capacity in your course? (4 responses received in total)

Themes recorded in instructor responses to T6	No. of responses reporting this theme/total responses received
Improving intuitiveness	3/4
Wider corpus of problems for class use	1/4

Table 10. Q4. Please include any other feedback that you would like to see to guide you on how you solved the problem. (40/80 students responded in total)

Themes recorded in student responses to Q4	No. of responses reporting this theme/total responses received
Explicit responses	15/40
Positive	6/15
Mixed positive leaning	3/15
Ambiguous	1/15
Mixed negative leaning	2/15
Negative	5/15
Feedback text was not helpful	7/40
Feedback text was not specific enough	7/40
Did not notice any notifications/feedback text	6/40
Suggestions on improving feedback	8/40
Improving UI design (eg: pointing to exact box)	4/8
Include guidance/link to material on how to fix errors	2/8
Provide hint button for stepwise hints	2/8

Students

Overall, the students liked the design decisions we made for the first iteration of our targeted feedback system (L14: 21.2% Strongly Agree, 35% Somewhat Agree; L15: 22.5% Strongly Agree, 26.2% Somewhat Agree; L16: 21.2% Strongly Agree, 37.5% Somewhat Agree; Figure 8), garnering mostly positive responses about the Notifications panel itself, usefulness of the interactive feedback, and the message texts. When asked to elaborate (Q4, Table 10), the positive responses stated the message texts and interactivity made it easy to find errors in equations. However, negative responses stated the messages were either not helpful or not specific enough, pointing out that the language of the messages was sometimes not clear enough, and that better indicators of where the errors occurred are needed. The exact points of errors in the student's solution are shown by interactive notifications such as a clickable text that highlights the box or the equation with an error, but the student still depends on the provided text message to know more details about what went wrong. In some situations, the notifications just showed "incorrect answer" instead of providing details, which may also be tied to students not seeing any feedback due to not noticing the panel at all. On the other hand, providing too much feedback about the error is considered bad by instructors (and is in fact their chief criticism). Either way, several suggestions were provided for improving the feedback through improving the texts and the interactivity, and future research is needed to find a proper middle ground regarding the right amount of feedback to provide.

Table 11. T2. How do you feel your students would react to using such a system in its current form as a supplemental tool to practice problems and get immediate feedback? (13 responses received in total)

Themes recorded in instructor responses to T2	No. of responses reporting this theme/total responses received
Students would not use the system unless mandated to do so	4/13
Students might be willing to try this	2/13
Few would actually use this (of their own volition)	3/13
Good students wouldn't need this	2/13
Initial learning hurdle might be deterrent	4/13

Instructors

T1 (Table 6) asked instructors to express their opinions about the system's impact on classes. Their responses indicate that despite the usefulness of the system and its immediate feedback, there were a couple of aspects that could interfere with learning (4 of 15). The system seemed a little too helpful in providing automated support for working with units of quantities and algebra, which might affect their skill levels as engineers later. Some instructors supposed that this would shift student focus to technical issues rather than the materials, and that solving by hand might be faster. One response to T1 mentioned that they would like more nuanced feedback, and we have improved on this in subsequent iterations.

When asked how they felt students would react to the system in its current form (T2, Table 11), instructors expressed concern about willing adoption by students. Attaching some kind of grading to the exercises solved would create an impetus for more students to use this. Direct and timely feedback on their solution attempts would make the system attractive and would be the primary reason for them to prefer this system. However, for more widespread acceptance, instructors feel that the interface needs to be more intuitive, with a lower learning curve, since the extra work of learning to use a new system together with the interface operations as they are now might deter students from using this system more extensively. Additionally, the repetitive nature of some of the tasks involved in setting up the system might turn away students. This aligns with opinions we saw from the student survey, and gives us future directions to improve the interface design.

Regarding adoption (T3, Table 12), instructors seemed more inclined to use the system as a supplementary tool or for low-stakes assessments in classes, as opposed to high-credit graded homework exercises and exams. Among the positives, instructors acknowledged the utility of the system in discouraging cheating and in the automated feedback provided as major positives for in-class use. To quote one of the instructor's responses, "[The] way it's set up it really discourages

Table 12. T3. With further development, in what ways could it be of value as a learning tool? For example, as a supplemental tutorial option, a homework mechanism, a testing tool with ability for awarding partial credit? (11 responses received in total)

Themes recorded in instructor responses to T3	No. of responses reporting this theme/total responses received
Use as a supplemental tutorial for sample problems	3/11
Use as an in-class teaching tool for increased engagement	3/11
Solving homework problems	5/11
(Explicit) Do not use for testing	2/11

just using a solution manual or copying someone else's work". This makes it great for a homework tool. However, chiefly, the instructors reasoned that the system provided too much support to the students (also seen in responses to T1, Table 6). They also responded they would like to see grading support for exercise problems first, including support for partial credit.

Among standout comments, we received two responses in T1 and T3 about students potentially gaming the system to get it to generate hints or accept correct answers without doing the work. One instructor commented, "It takes most of the thinking out of the problem solving. Instead students can, and likely will, quickly learn to game the system, since all possible equations are just given to them." This also shows up in T3, with one response stating, "It gives entirely too much assistance to the students, and they can quickly learn to game [the] system to give them correct answers, without actually understanding any of the material." This brings up an interesting point regarding the interface design, that making the system robust against potential misuse is an important avenue to consider when improving our system.

DISCUSSION

Our surveys of students and instructors report overall positive responses regarding our design choices and demonstrate success in achieving the original system goals. Some criticisms we plan to address in future versions of our system. Responses are summarized in the following sections.

Summary of positive responses

Traditional methods of solving problems manually using pen, paper and calculators have an associated algebraic tedium that distracts focus from practicing topics learned in class. Our interface was designed to reduce this tedium and instead focus on connecting concepts with class content. Our interface uses a modular approach to construct solutions to problems using equation templates drawn from palettes. Both instructors and students appreciated these features for constructing common relationships, automatic unit computations, and to rapidly solve large systems of equations. Instructors and students liked the similarity to traditional pen and paper solutions, which confirms the findings of Pierce and Stacey (Pierce and Stacey 2001) that providing an experience similar to solving problems on pen-and-paper and working closely with instructors can improve adoption.

Instructors reported the system to be easy to learn, requiring only 1-2 practice problems or 20-30 minutes of active engagement to come up to speed. Students reported the interface interactions with visual elements were easy to understand, indicating that both the design of the interface and the tutorial materials provided were effective in guiding. Instructors welcomed the targeted feedback system, although the students' and instructors' opinions were divided on the helpfulness and specificity of the feedback provided, with students requesting more specific feedback and instructors suggesting that the feedback provided might be too much.

Summary of criticisms and directions for improvement

While students and instructors agreed that the feel of the system was similar to pen and paper and reduced algebraic tedium, some of the interface features created additional tedium. The same workflow for creating and solving complex systems of equations that students liked appeared tedious when working with simple equations. This difficulty can partly be attributed to a lack of familiarity with the system. Despite the ease of learning to use the system, instructors were concerned that the learning curve might deter students unless the course explicitly required them to use the system. Particularly for educational systems where users are not intended to use it in the long-term, systems should be more self-descriptive, or at least provide detailed overviews to easily familiarize students with the interface (Oppermann 2002). This is a great motivation for us to find ways to reduce the training time by increasing intuitiveness in general and improving the tutorial to comply with widely accepted usability standards (Dzida 1996), which in turn can increase adoption and acceptance. Both instructors and students also requested greater flexibility in creating systems of equations.

Students also critiqued interface design features that affected readability. The vertical layout of elements required a lot of scrolling back and forth to access commonly used elements (equations, workspaces, problem prose), and viewing the interface in small screens typical to the laptops used by students resulted in visual elements obscuring each other. While students reported the interface interactions were intuitive, they also noted difficulties, which can be attributed to both unfamiliarity with the system and a mismatch of expectations of how the interface should interact. Our original motivations were to simplify interactions through simple click-based shortcuts. This seems to have added cognitive load. Research shows that good interfaces should provide means for users to transition from easy-to-learn but inefficient methods of novices to difficult-to-learn but efficient methods of experts (Lane et al. 2005), and a good way to ensure ease of use is to increase visibility of elements (Galitz 1997). Student suggestions for improving the interface design include collapsible workspaces, and a more menu-driven workflow.

Instructors and students generally liked the feedback system, but their opinions on the purpose of feedback were divided. Students asked for greater clarity, while instructors were worried about the system providing feedback that is too detailed. We note our original system design was motivated by the delay in providing feedback and inconsistent quality and helpfulness of the feedback in traditional settings, which is a major issue in large engineering classes (Hounsell et al. 2005). We also note that the system at the time of writing did not support semantic correctness of the solution in the context of the exercise problem. Checking for semantic correctness involves analyzing the system of equations that led to the solution, in comparison to the model solution constructed by the instructor. Examples include identifying errors in the equation setup at a granular level, such as incorrect parameters used, incorrect signs used, incorrect setup of associations, etc. Such feedback has since been incorporated. The feedback provided should be mindful of the instructors' concerns about providing too much help. At the same time, the system must be suitable for both teaching and student assistance, as well as be robust enough to apply to several different problems without intervention from the instructor or a teaching assistant. Given the benefits of using compact and targeted feedback over repetitive feedback (Di Eugenio et al. 2008), we could leverage user interface interactions by using visual cues of error locations alongside interactive text to provide a rich, compact, text-based feedback. Additionally, demonstrating positives in performance expectancy and effort expectancy as per the UTAUT model (Venkatesh et al. 2003) can gain confidence of students and instructors alike, improving adoption of our system.

FUTURE WORK

We see several avenues for improvement. Most important are these aspects of the interface design:

- More flexible equation input. We are in the process of redesigning the arithmetic equation templates for increased flexibility in creating basic arithmetic relationships, which would help eliminate some of the algebraic tedium that students reported regarding the problem-solving process.
- Redesigning the interface to better accommodate larger visual elements and require less scrolling. Possible changes include an increased workspace area, tabbed or collapsible workspaces, an easily accessible equation bank hidden behind a menu or a collapsible sidebar, and a clearly visible notifications panel with interactive text to point to error locations.
- Reconsider the pros and cons of click-and-drop versus drag-and-drop for manipulating the interface elements.

The most important innovations of our system will come from its ability to provide appropriate feedback to students about their errors. In the prototype described in this paper, we only supported syntactical feedback that provided guidance on mathematical consistency, outside of the context of the solution to the exercise problem. Since then, we have added semantic analysis that looks at the system of equations in the light of a model solution constructed by the instructor. The feedback generated by this is more targeted, and preliminary user studies have shown benefits of this in helping students understand specific errors to find correct solutions. Additionally, the mechanism for providing semantic feedback can also be leveraged to provide additional features such as adding instructor-tunable feedback, where the instructor can decide how much feedback to provide depending on the setting in which the exercise is provided; and a fair grading mechanism that accounts for a student's attempt together with their submitted answer.

Finally, we want to explore the ability to award grades based on problem-solving attempts. Currently, the system supports all/nothing grading, where students must submit the right quantity and unit combination to receive points for their attempt. However, this leaves room for a false positive and a false negative scenario. A false positive can occur when a student somehow contrived a correct solution and games the interface to enter this quantity into the submission box and get full credit without using the right equations. Grading should include analysis of the steps given toward the answer. A false negative can occur when a student gets most of the equations correct and has the right approach, but due to an incorrect setup they might not be able to compute their answer correctly. A common scenario occurs when a student computes an answer with the wrong sign at the end, while still having the right approach. In both cases, one would need to identify the equations and parameters used. If they match what the instructor expected them to submit, then the system should offer partial credit for it accordingly. Our future work includes both a careful analysis of instructors' feedback of a sample of student homework submissions to compare against our system's feedback, and a new feedback engine that can capture the semantic differences between student answers and reference answers, thus supporting correct but different approaches. We also hope to give instructors control over the feedback that the system provides.

REFERENCES

Ainsworth, Shaaron. 2008. "The Educational Value of Multiple-Representations When Learning Complex Scientific Concepts." Visualization: Theory and Practice in Science Education, 191–208. https://doi.org/10.1007/978-1-4020-5267-5_9.

Anderson, John R, Albert T Corbett, Kenneth R Koedinger, and Ray Pelletier. 1995. "Cognitive Tutors: Lessons Learned." The Journal of the Learning Sciences 4 (2): 167–207.

Azevedo, Roger, and Dragan Gašević. 2019. "Analyzing Multimodal Multichannel Data about Self-Regulated Learning with Advanced Learning Technologies: Issues and Challenges." Computers in Human Behavior 96 (July):207–10. https://doi.org/10.1016/J.CHB.2019.03.025.

Chen, John C., Dexter C. Whittinghill, and Jennifer A. Kadlowec. 2010. "Classes That Click: Fast, Rich Feedback to Enhance Student Learning and Satisfaction." Journal of Engineering Education 99 (2): 159–68. https://doi.org/10.1002/j.2168-9830.2010.tb01052.x.

Chi, Min, Kurt Vanlehn, Diane Litman, and Pamela Jordan. 2010. "Inducing Effective Pedagogical Strategies Using Learning Context Features." In Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 6075 LNCS:147–58. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-13470-8_15.

Di Eugenio, Barbara, Davide Fossati, Susan Haller, Dan Yu, and Michael Glass. 2008. "Be Brief, And They Shall Learn: Generating Concise Language Feedback for a Computer Tutor." International Journal of Artificial Intelligence in Education 18 (4): 317–45.

Dollár, Anna, and Paul S. Steif. 2008. "An Interactive, Cognitively Informed, Web-Based Statics Course." International Journal of Engineering Education 24 (6).

Dori, Yehudit Judy, and John Belcher. 2005. "How Does Technology-Enabled Active Learning Affect Undergraduate Students' Understanding of Electromagnetism Concepts?" The Journal of the Learning Sciences 14 (2): 243–79. https://doi.org/10.1207/S15327809JLS1402_3.

Drijvers, Paul. 2015. "Digital Technology in Mathematics Education: Why It Works (Or Doesn't)." In Selected Regular Lectures from the 12th International Congress on Mathematical Education, 135–51. Springer, Cham. https://doi.org/10.1007/978-3-319-17187-6_8.

Dzida, Wolfgang. 1996. "International Usability Standards." ACM Computing Surveys 28 (1). https://doi.org/10.1145/234313.234386.

Ertmer, Peggy A., and Anne Ottenbreit-Leftwich. 2013. "Removing Obstacles to the Pedagogical Changes Required by Jonassen's Vision of Authentic Technology-Enabled Learning." Computers & Education 64 (May):175–82. https://doi.org/10.1016/J.COMPEDU.2012.10.008.

Galitz, Wilbert O. 1997. The Essential Guide to User Interface Design. SIGCHI Bulletin. Vol. 29.

Georghiades, Petros. 2000. "Beyond Conceptual Change Learning in Science Education: Focusing on Transfer, Durability and Metacognition." Educational Research 42 (2): 119–39. https://doi.org/10.1080/001318800363773.

Halverson, Lisa R., and Charles R. Graham. 2019. "Learner Engagement in Blended Learning Environments: A Conceptual Framework." Online Learning 23 (2): 145–78. https://doi.org/10.24059/olj.v23i2.1481.

Hounsell, Dai, Jenny Hounsell, Judith Litjens, and Velda McCune. 2005. "Enhancing Guidance and Feedback to Students: Findings on the Impact of Evidence-Informed Initiatives." European Association for Research on Learning and Instruction (EARLI) 11th Biennial Conference, Nicosia, Cyprus, no. January 2001.

Ioannou, Andri, Skevi Demetriou, and Maria Mama. 2014. "Exploring Factors Influencing Collaborative Knowledge Construction in Online Discussions: Student Facilitation and Quality of Initial Postings." American Journal of Distance Education 28 (3): 183–95. https://doi.org/10.1080/08923647.2014.926780.

Johnstone, A. H. 1991. "Why Is Science Difficult to Learn? Things Are Seldom What They Seem." Journal of Computer Assisted Learning 7 (2): 75–83. https://doi.org/10.1111/J.1365-2729.1991.TB00230.X.

Kirn, Adam, and Lisa Benson. 2018. "Engineering Students' Perceptions of Problem Solving and Their Future." Journal of Engineering Education 107 (1): 87–112. https://doi.org/10.1002/jee.20190.

Koohang, Alex, Paliszkiewicz Joanna, Jerzy Gołuchowski, and Jeretta Horn Nord. 2016. "Active Learning for Knowledge Construction in E-Learning: A Replication Study." Journal of Computer Information Systems 56 (3): 238–43. https://doi.org/10.1080/08874417.2016.1153914.

Lane, David M, H Albert Napier, S Camille Peres, Aniko Sandor, H Albert, Napier S Camille Peres, and Anikó Sándor. 2005. "Hidden Costs of Graphical User Interfaces: Failure to Make the Transition from Menus and Icon Toolbars to Keyboard Shortcuts." International Journal of Human-Computer Interaction 18 (2): 133–44. https://doi.org/10.1207/S15327590IJHC1802_1.

Lemke, Jay L. 1997. "Cognition, Context, and Learning: A Social Semiotic Perspective." Situated Cognition, December, 37–55. https://doi.org/10.4324/9781003064121-3.

Litzinger, Thomas A., Peggy Vanmeter, Carla M. Firetto, Lucas J. Passmore, Christine B. Masters, Stephen R. Turns, Gary L. Gray, Francesco Costanzo, and Sarah E. Zappe. 2010. "A Cognitive Study of Problem Solving in Statics." Journal of Engineering Education 99 (4). https://doi.org/10.1002/j.2168-9830.2010.tb01067.x.

Lund, Andreas, Eiliv Hauge, and Trond Eiliv Hauge. 2011. "Designs for Teaching and Learning in Technology-Rich Learning Environments." Nordic Journal of Digital Literacy 6 (4): 258–71. https://doi.org/10.18261/ISSN1891-943X-2011-04-05.

Montfort, Devlin, Geoffrey L. Herman, Shane Brown, Holly M. Matusovich, Ruth A. Streveler, and Olusola Adesope. 2015. "Patterns of Student Conceptual Understanding across Engineering Content Areas." The International Journal of Engineering Education. 31 (6): 1587–1604.

Nelson, Katherine G., Ann F. McKenna, Sarah K. Brem, Jonathan Hilpert, Jenefer Husman, and Eva Pettinato. 2017. "Students' Misconceptions about Semiconductors and Use of Knowledge in Simulations." Journal of Engineering Education 106 (2): 218–44. https://doi.org/10.1002/JEE.20163.

Oppermann, Reinhard. 2002. "User-Interface Design." Handbook on Information Technologies for Education and Training, 233–48.

Pierce, Robyn, and Kaye Stacey. 2001. "Observations on Students' Responses to Learning in a CAS Environment." Mathematics Education Research Journal 2001 13:1 13 (1): 28–46. https://doi.org/10.1007/BF03217097.

Pitterson, Nicole. 2015. "Undergraduate Engineering Students' Understanding of Complex Circuit Concepts: An Investigation of the Intersection of Students' Prior Knowledge, Design of Learning Environments and the Nature of the Content." Open Access Dissertations, January.

Popovici, Anca, and Cosmina Mironov. 2015. "Students' Perception on Using eLearning Technologies." Procedia - Social and Behavioral Sciences 180 (May):1514–19. https://doi.org/10.1016/J.SBSPRO.2015.02.300.

Reinholz, Daniel. 2016. "The Assessment Cycle: A Model for Learning through Peer Assessment." Assessment and Evaluation in Higher Education 41 (2): 301–15. https://doi.org/10.1080/02602938.2015.1008982.

Rogers, Everett M. 1995. "Diffusion of Innovations (4th Ed.) Free Press." New York.

Sands, David, Mark Parker, Holly Hedgeland, Sally Jordan, and Ross Galloway. 2018. "Using Concept Inventories to Measure Understanding." Higher Education Pedagogies 3 (1): 173–82. https://doi.org/10.1080/23752696.2018.1433546.

Shaffer, Clifford A, Thomas L Naps, and Eric Fouh. 2011. "Truly Interactive Textbooks for Computer Science Education." In Proceedings of the Sixth Program Visualization Workshop, 97–103.

Shaffer, Clifford A., Ville Karavirta, Ari Korhonen, and Thomas L. Naps. 2011. "OpenDSA: Beginning a Community Active-EBook Project." Proceedings - 11th Koli Calling International Conference on Computing Education Research, Koli Calling'11, 112–17. https://doi.org/10.1145/2094131.2094154.

Steif, Paul S. 2004. "An Articulation of the Concepts and Skills Which Underlie Engineering Statics." In 34th Annual Frontiers in Education, 2004. FIE 2004., 559–64. IEEE. https://doi.org/10.1109/FIE.2004.1408579.

Steif, Paul S., and Anna Dollár. 2005. "Reinventing the Teaching of Statics." International Journal of Engineering Education 21 (4 PART I AND II). https://doi.org/10.18260/1-2--13940.

Steif, Paul S., and Anna Dollár. 2009. "Study of Usage Patterns and Learning Gains in a Web–based Interactive Static Course." Journal of Engineering Education 98 (4): 321–33. https://doi.org/10.1002/j.2168-9830.2009.tb01030.x.

Steif, Paul S., and John A. Dantzler. 2005. "A Statics Concept Inventory: Development and Psychometric Analysis." Journal of Engineering Education 94 (4): 363–71. https://doi.org/10.1002/j.2168-9830.2005.tb00864.x.

Streveler, Ruth A., Thomas A. Litzinger, Ronald L. Miller, and Paul S. Steif. 2008. "Learning Conceptual Knowledge in the Engineering Sciences: Overview and Future Research Directions." Journal of Engineering Education 97 (3): 279–94. https://doi.org/10.1002/J.2168-9830.2008.TB00979.X.

Van Den Boom, Gerard, Fred Paas, Jeroen J.G. Van Merriënboer, and Tamara Van Gog. 2004. "Reflection Prompts and Tutor Feedback in a Web-Based Learning Environment: Effects on Students' Self-Regulated Learning Competence." Computers in Human Behavior 20 (4): 551–67. https://doi.org/10.1016/J.CHB.2003.10.001.

Venkatesh, Morris, Davis, and Davis. 2003. "User Acceptance of Information Technology: Toward a Unified View." MIS Quarterly 27 (3): 425. https://doi.org/10.2307/30036540.

Watson, Mary Katherine, Joshua Pelkey, Caroline R. Noyes, and Michael O. Rodgers. 2016. "Assessing Conceptual Knowledge Using Three Concept Map Scoring Methods." Journal of Engineering Education 105 (1): 118–46. https://doi.org/10.1002/JEE.20111.

Wiliam, Dylan, and Marnie Thompson. 2017. "Integrating Assessment with Learning: What Will It Take to Make It Work?" In The Future of Assessment: Shaping Teaching and Learning, 53–82. Taylor and Francis. https://doi.org/10.4324/9781315086545-3.

Yang, Dazhi, Ruth Streveler, Ronald L. Miller, Inanc Senocak, and Jim Slotta. 2020. "Using Schema Training to Facilitate Students' Understanding of Challenging Engineering Concepts in Heat Transfer and Thermodynamics." Journal of Engineering Education 109 (4): 743–59. https://doi.org/10.1002/JEE.20360.

AUTHORS

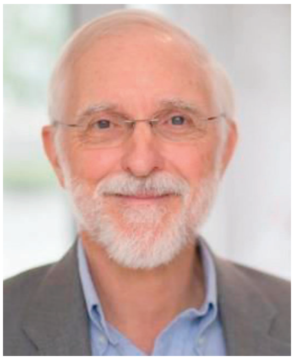
Arinjoy Basak is a PhD candidate in the Department of Computer Science at the Virginia Polytechnic Institute and State University, advised by Prof. Clifford A. Shaffer. He obtained his Bachelors in Computer Science from the Indian Institute of Engineering Science and Technology, Shibpur in 2016, and his Masters in Computer Science from Virginia Tech in 2019. His research interests include digital education, targeted feedback generation, and understanding problem-solving processes.

Nicole P. Pitterson is an Assistant Professor in the Department of Engineering Education at Virginia Tech. She holds a B.Sc in Electrical and Electronic Engineering from the University of Technology, Jamaica, a M.Sc. in Manufacturing Engineering from Western Illinois University, and a Ph.D. in Engineering Education from Purdue University. Upon completion of her doctoral degree, Dr. Pitterson worked as a postdoctoral research scholar at Oregon State University. Her research interests are exploring students' disciplinary identity through engagement with knowledge, curriculum design, assessment and evaluation and teaching for conceptual understanding. Dr. Pitterson is dedicated to bridging the gap between theoretical concepts with practical applications. She also aims to guide students to develop critical understanding of core engineering concepts that goes beyond rote memorization so that they can adapt to the changing demands of a global workforce.

Clifford A. Shaffer is Professor of Computer Science at Virginia Tech, where he has been since 1987. He received his PhD from the University of Maryland in 1986. Dr. Shaffer's work related to Algorithm Visualization (AV) dates to the mid-1990s, when he oversaw the development of the SWAN AV system. In 2006 he initiated the AlgoViz project, which cataloged a wide variety of AV systems with the goal to support increased use of AVs by CS educators. This effort led in turn to the OpenDSA project, an international effort to develop infrastructure and content for creating eTextbooks for

Computer Science courses. OpenDSA began in 2011 and is now being used by thousands of students per year at universities in several countries. Dr. Shaffer is an IEEE Senior Member, and an ACM Distinguished Educator.

Sneha Patel Davison is an instructor in the Biomedical Engineering and Mechanics Department at Virginia Tech. She earned her Bachelors of Science, her Masters of Science, and her Doctoral degree from the Engineering Mechanics department also at Virginia Tech. Her research interests include exploring the most effective methods to teach students introductory level mechanics, especially in the large classroom environment.



David Dillard is the Adhesive and Sealant Science Professor in the Biomedical Engineering and Mechanics Department at Virginia Tech. He has worked extensively in the field of adhesive bonding, having experience in structural adhesives for aerospace, automotive, and infrastructure applications; adhesives and coatings for microelectronic applications; pressure sensitive adhesives; elastomeric adhesives and sealants; and polymeric membranes. He has authored or co-authored over 210 refereed publications and regularly teaches courses in adhesion science, polymer viscoelasticity, and sustainable energy solutions. With four decades of experience as an educator, he is interested in new ways to leverage technology to have a positive impact on student learning and assessment.

Jacob Grohs is an associate professor in the Department of Engineering Education and Interim Director of the Center for Educational Networks and Impacts in the Institute for Creativity, Arts, and Technology at Virginia Tech. His primary research interests focus on systems thinking, applied educational research through multi-stakeholder partnerships, and collaborative change.