Enhancing Online Problems Through Instructor-Centered Tools for Randomized Experiments

Joseph Jay Williams
National University of Singapore
williams@comp.nus.edu.sg

Anna N. Rafferty
Carleton College
Northfield, MN, USA
arafferty@carleton.edu

Dustin Tingley
Harvard University
Cambridge, MA, USA
dtingley@gov.harvard.edu

Andrew Ang
Harvard University
Cambridge, MA, USA
andrew_ang@.harvard.edu

Walter S. Lasecki
University of Michigan
Ann Arbor, MI, USA
wlasecki@umich.edu

Juho Kim
KAIST
Daejeon, South Korea
juhokim@cs.kaist.ac.kr

ABSTRACT

Digital educational resources could enable the use of randomized experiments to answer pedagogical questions that instructors care about, taking academic research out of the laboratory and into the classroom. We take an instructor-centered approach to designing tools for experimentation that lower the barriers for instructors to conduct experiments. We explore this approach through DynamicProblem, a proof-of-concept system for experimentation on components of digital problems, which provides interfaces for authoring of experiments on explanations, hints, feedback messages, and learning tips. To rapidly turn data from experiments into practical improvements, the system uses an interpretable machine learning algorithm to analyze students' ratings of which conditions are helpful, and present conditions to future students in proportion to the evidence they are higher rated. We evaluated the system by collaboratively deploying experiments in the courses of three mathematics instructors. They reported benefits in reflecting on their pedagogy, and having a new method for improving online problems for future students.

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation: Misc.

Author Keywords

Online education; MOOCs; randomized experiments; dynamic experimentation; instructional design

INTRODUCTION

Online problems are widely used to assess students' knowledge and to provide feedback about whether and why their answers are correct, in settings from on-campus learning management systems to Massive Open Online Courses (MOOCs). Instructors can improve student learning by incorporating explanations, hints, and pedagogical tips into online problems [10], just as occurs in face-to-face tutoring. However, when instructors give face-to-face explanations, they naturally vary the explanations that they give, enabling them to compare which explanations students find helpful and which leave them confused based on students' body language and verbal responses. This adaptation is rare with online problems: instructors typically create one feedback message for a problem, and that message is provided to every student who interacts with the problem. This eliminates the opportunity to iteratively improve the message over time, and the opportunity for the instructor to reflect on what types of messages are most helpful for students.

One way to address this challenge is through experimentation in online problems: randomized experiments can be used to compare alternative versions of explanations, and data can be collected about how helpful students find each explanation. This affordance of online educational resources is increasingly being used by researchers in psychology and education to move randomized experiments from the laboratory into real-world courses [34].

While these experiments are deployed in some instructors' classes, these experiments often have limited impact on teachers' pedagogy and may not lead to persistent changes in courses that directly impact students. Even if the instructor and researcher are collaborating on an experiment by virtue of the experiment being conducted in the instructor's course, the researcher's needs and perspective are likely to dominate. Researchers have more experience conducting experiments and are the ones doing the work of programming randomization, implementing conditions, and managing data logging and analysis. Instructors might only act as gatekeepers for approving a deployment of experiments, rather than as stakeholders in the design and recipients of the benefits. This could also be a lost opportunity for academic research, as instructors could improve experiments by contributing their teaching experience, and help scientists rethink the opportunities for experiments to more directly inform instructional design and produce concrete curricula changes that have practical benefits for students.

CHI 2018, April 21–26, 2018, Montreal, QC, Canada

ISBN 978-1-4503-5620-6/18/04…15.00

DOI: https://doi.org/10.1145/3173574.3173781

This paper therefore takes an *instructor*-centered approach to designing tools for experimentation. Based on observations from deploying field experiments with instructors, our goal is to reduce the programming knowledge and time required for instructors to conduct experiments, and to accelerate the use of data from experiments to enhance curricula materials and help future students.

We instantiate guidelines for instructor-centered experimentation in *DynamicProblem*, a proof-of-concept system for instructors to work with a researcher to experiment on instructional components of interactive math problems. Experiments compare different *elaboration messages*, alternative versions of explanations, hints, feedback messages, and learning tips. The system, *DynamicProblem*, directly integrates the design of experiments into instructors' authoring of instructional content. To accelerate the use of data from experiments to enhance curricula materials, the system uses an interpretable machine learning algorithm to automatically analyze students' subjective ratings and present the most effective conditions to future students.

The first author worked with three university instructors in using DynamicProblem to compare alternative elaboration messages in problems in their courses. Instructors reported that the system successfully reduced the programming experience and time needed to conduct experiments, that designing experiments helped instructors reflect on pedagogy, and that instructors believed dynamic experimentation would enable practical improvement of curricula materials by providing the best problem components to future students.

In summary, the contributions of this paper are:

- Guidelines for an instructor-centered approach to designing tools for experimentation: Reducing the programming and time requirements for instructors to conduct experiments, and shortening the pipeline for data to be used to produce practical benefits for students.
- An instantiation of this instructor-centered approach in a proof-of-concept system: DynamicProblem is an end-user tool for conducting experiments on components of a problem – hints, explanations, and learning tips – which uses machine learning to automatically analyze data and provide the highest rated components to future students.
- Presentation of case studies from field deployments of DynamicProblem in the university courses of three instructors.

RELATED WORK

We consider related work on technology that enables experimentation in digital educational resources, efforts to involve instructors in research, and the specific opportunities for experimentation on components of online math problems.

Technology for Experimentation

One barrier to using experimental studies to investigate possible enhancements to digital educational resources is the lack of tools for end-user experimentation by instructors and learning researchers. Ideally, instructors could be involved in the "democratization of data science" [14], analogous to how [22] help Reddit community members conduct experiments with moderation and community policies.

As "A/B testing" for user interfaces and marketing becomes increasingly widespread in the technology industry (e.g., see [17]), tools like Optimizely and Google Content Experiments have emerged to enable end-user experimentation on websites. However, most widely used Learning Management Systems (Blackboard, Canvas, Moodle) and MOOC platforms (Coursera, NovoEd) do not provide instructor-facing tools for experimentation. The security and data privacy considerations of educational platforms pose obstacles to adding custom code and using industry tools for generic website experimentation. Plugins often require the use of specific standards like Learning Tools Interoperability [23].

Some platforms (such as edX and ASSISTments [13]) are introducing tools that make it easier for instructors and researchers to run experiments without writing code, although these efforts are currently in the minority. Moreover, these tools require instructors to do the work of linking linkdo not automatically link students' the condition students are typically require extensive work in linking data about students' assignment to conditions with data about student outcomes [25].Critically, tools designed for experimentation do not encourage experiments to be used directly for course improvement, such as providing data rapidly *while* a course is running, or providing automatic methods for transitioning from random assignment to providing the single best condition.

Involving Instructors in Research

There have been a number of efforts to involve instructors in the process of academic research, recognizing that simply disseminating the findings of research studies may often fail to change instructors' teaching practices [4]. These have largely focused on research using qualitative [24], design [3], and data mining/analytics methods [21], rather than student-level randomized experiments, which are traditionally difficult to conduct in classes. *Action research* in education (e.g., [24]) and HCI [12] proposes that instructors solve pedagogical problems by taking a research-oriented approach, in considering relevant literature, formulating questions and hypotheses, and collecting (largely) qualitative data about what works in their own classrooms.

Design-based research describes a family of approaches by learning scientists [3, 7, 2] that emphasize development of theories by iterative development of interventions in realistic contexts. This focus leads to natural involvement of teachers as partners, such as in successfully co-designing digital assessments (e.g., [26, 36]). Both Design-Based Implementation Research [26] and Networked Improvement Communities [8] target teacher-researcher partnerships that focus on having research problems surfaced by teachers, as well as closing the loop by putting insights from research into practice. Similarly, the Open Learning Initiative [21] connected teams of instructors and cross-disciplinary researchers around collaborative design of online courses and analysis of assessment data.

Instructional Support in Interactive Math Problems

This paper explores instructor-centered experimentation in online math problems, as math instructors frequently use these problems for assessment and improving students' learning. Problems that incorporate high quality instructional scaffolding can promote learning far more than lectures or videos [16]. Elaboration messages have been specifically targeted in interactive math problems, with enhancement of problems through appropriate design of hints [15], explanations [35], and learning tips [6]. Despite the extensive research literature showing that high quality support *can* benefit learning, there are many open questions about how to provide the best instructional support and feedback in interactive problems (see [31] for a review), and a great deal of work to be done in testing the extent to which these broad principles apply to specific real-world courses and contexts.

CHALLENGES FOR INSTRUCTOR EXPERIMENTATION

We consider the challenges instructors face in three stages of conducting an experiment: (1) *Authoring & Deployment*: Creating and randomly assigning conditions– alternative versions of educational resources; (2) *Obtaining Data*: Linking data about student responses to the experimental conditions they received; (3) *Using Data*: Improving course content for future students based on data from an experiment.

This section elaborates on each stage of experimentation, presenting: (1) An illustrative example, drawn from a collaborative experiment between the first author and an instructor teaching an online course; (2) Challenges instructors face, identified from field observations and interviews with instructors, as described below; (3) Guidelines for designing instructor-centered tools for experimentation.

Field Observations and Interviews. We identified challenges instructors face in experimentation by drawing on observations from two sources. First, field observations from working with instructors to deploy experiments, which provided insight into the challenges that arose from real-world deployments with students. Second, interviews with instructors who had not yet run experiments, to understand their expectations and concerns about conducting experiments in their course. The field observations arise from the first author's experience while working with over 20 instructors and instructional designers to conduct 15 experiments in their courses between 2012-2016. These instructors taught both university courses and MOOCs in mathematics/STEM; the technology included platforms for on-campus Learning Management Systems (Moodle, Canvas), MOOCs (edX, NovoEd), and mathematics problems (the Khan Academy, ASSISTments); the experiments varied interactive problems, videos, webpage content, and emails; the topics of investigation ranged over cognitive, educational, and social psychology questions about motivation and problem-solving. The interviews with people who had not yet run experiments were 30-60 minute one-on-one discussions with ten instructors and instructional designers, who were teaching university-level courses that could use mathematics/STEM problems. The interviews explored instructors' beliefs about how experiments could be relevant to improving their courses, and what barriers and benefits they anticipated in working with researchers to conduct experiments.

1. Authoring & Deployment

Instructors need to author multiple conditions (versions of an educational resource) and randomly assign these to students.

Example. The instructor and researcher (the first author) investigated how students' reflection while watching videos could be enhanced, by adding different open-ended question prompts below a video. Most course platforms do not provide a means for random assignment, so the researcher embedded Javascript code in a course webpage to present different reflective questions (note that some platforms do not custom scripting). To ensure a student's assignment to a question could be linked to their later behavior in the course, students were assigned to conditions (questions) based on the value of the last digit of a hashed numerical identifier from the platform (making the assumption that distribution of these digits is random). The instructor could not use the course authoring tools to edit or preview the questions/conditions, as these were stored directly in JavaScript. Back and forth iterations on authoring/editing therefore had to take place over email and in Google Docs, and the instructor could not directly preview student experience in different conditions.

Challenges. The field observations and interviews suggested that instructors are not aware of how to author and deploy experiments in their course platforms without assistance from programmers. Most instructors lack programming skills (or end-user tools) for authoring and implementing randomized assignment of alternative versions of course content.

Instructors were therefore not very involved in the details of designing experiments in their courses. Instructors had high level verbal discussions of experiments, but researchers or programmers implemented the actual conditions in code divorced from course authoring tools. The delays in back-and-forth communication reduced instructor engagementt and agency in authoring conditions/content for experiments.

Design Guidelines. Minimize the programming knowledge and time needed for instructors to author alternative versions of course content, and directly deploy randomized content to their students.

2. Obtaining Data

Instructors need to obtain data about how student outcomes/behaviors are impacted by the condition to which students were randomly assigned.

Example. To analyze the results, the researcher had to obtain log data from the platform– about students' viewing time of subsequent videos, and accuracy on future problems. These dependent variables for each student than had to be linked to the question condition that each student was assigned to. This required connecting anonymous identifiers associated with log data to the different identifiers that were available to the Javascript code for randomization, by obtaining a mapping from the platform. Figuring out this process was sufficiently complicated that data was not obtained until a month after the experiment was deployed and data collected.

Challenges. Instructors lack knowledge about how to obtain student data and link it to the course content to which students were randomly assigned. Even technology-savvy researchers face technical barriers and long delays in obtaining data from educational platforms and linking it to earlier randomized content. This often depends on idiosyncrasies and different standards for the separate processes of data logging and random assignment to course content, as well as knowledge of tools for munging data from different sources.

Design Guidelines. Help instructors obtain student responses grouped by experimental condition, with minimal time spent figuring out how to combine different sources of data.

3. Using Data: Improving resources for future students

As instructors obtain data about the effect of experimental conditions on the students who have participated so far, this data can inform their decisions about which conditions/resources to present to subsequent students.

Example. Because the researcher had to manually link data about student outcomes to condition assignment, the instructor did not receive the data in a useful format until the course was completed. Upon obtaining data, the instructor deliberated whether to continue the experiment in the next offering of the course, or terminate the experiment by choosing one condition/question for all future students. Although the data was not conclusive about which condition/question was best, the instructor decided to avoid the substantial overhead of conducting another experiment in the next course run. The code enabling random assignment was stripped out of the course, and the course platform used to re-author a single condition/question. It should be noted that the instructor was aware they were giving all future students a condition/question that may not have actually been the best. There was only inconclusive evidence in the form of a small numerical trend, but no statistically significant or even marginal result.

Although the instructor wanted to make a decision that was better founded in the data, they were concerned that even if the researcher could obtain data more rapidly in the next course iteration, the instructor would have to deal with the mental overhead of continually examining that data and determining when they were justified in ending the experiment. Moreover, even in the best case scenario where data from early students identified which condition would be best for later students in the same course, making the change to present that condition to everyone could introduce bugs, as it required stripping out the code and re-authoring content in the course platform.

Challenges. Instructors were often unsure of whether and how data from experiments will help them change their instructional practice, and did not see how traditional scientific experiments would immediately benefit their students. They had the impression (and experience) that obtaining and analyzing data from experiments would only be completed well after their course was finished. It wasn't clear to them what the exact pathway was from conducting an experiment to changing course materials for future students, and they expected that it would involve additional time, mental overhead, programming effort, and risk of bugs or negative student experience.

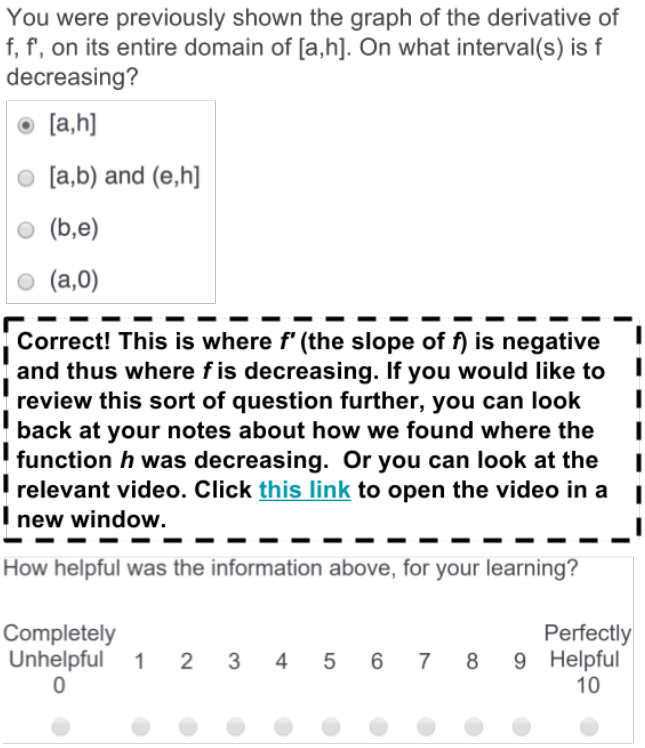

Figure 1. Typical student interaction with a math problem (selected from case study 3). After students submit an answer to an online math problem, one of the set of alternative elaboration messages (experimental conditions) is assigned to them, shown in the dotted box. The student is asked to rate how helpful the message is for their learning. The algorithm for dynamic experimentation uses this data to reweight randomization and present future students with higher rated elaboration messages, and the instructor can view this data as well. In this case, the student was assigned to the message that contained a learning tip about reviewing relevant materials. The corresponding instructor interface for authoring these messages is shown in Figure 2, which shows that this message was the second condition.

Design Guidelines. Reduce the delays and costs to instructors in using data from each student's participation in an experiment, in order to make practical improvements to educational resources that benefit future students.

DYNAMICPROBLEM SYSTEM

We present a proof-of-concept system, called *DynamicProblem*, that helps instructors conduct experiments on instructional components of problems that we call *elaboration messages*, such as explanations for correct answers [28], hints [27], and learning tips [6]. The system exemplifies our broader instructor-centered approach, lowering the overhead associated with conducting experiments and enabling real-time analysis of data so that future students are assigned to the conditions that were effective for past students.

DynamicProblem can be plugged into any learning management system or MOOC platform that supports the ubiquitous Learning Tools Interoperability (LTI) standard [23]. This includes major platforms like Canvas, Blackboard, Moodle, edX, Coursera. (Instructions for how any instructor can use DynamicProblem are available at www.josephjaywilliams.com/dynamicproblem, along with the

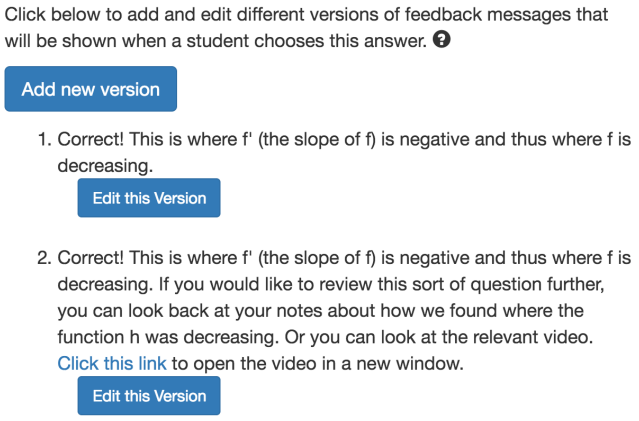

Figure 2. Interface for instructors and researchers to author elaboration messages (experimental conditions) in problems. For illustration, this shows what the instructor for Case Study 3 saw, in their experiment comparing their existing correctness feedback to correctness feedback plus a learning tip. Figure 1 shows the interface of a student who participated in this experiment and received the second condition. The link to View Data and Policy Dashboard navigates to the table in Figure 3.

open sourced code). Students navigating to a DynamicProblem page simply see a problem, choose an answer, and then receive an elaboration message. Students are prompted to rate how helpful the elaboration message is, on a scale from 0 (completely unhelpful) to 10 (perfectly helpful). Figure 1 illustrates a student's interaction with a problem from Case Study 3.

When instructors and researchers go to the same DynamicProblem page, they access a composite interface that provides access to pages for authoring and previewing problems, as well as for designing, deploying and examining data from experiments on elaboration messages. In addition to a page for authoring the text of the problem's question and answer choices (standard in Learning Management Systems), Figure 2 shows the interface for instructors and researchers to add, review, and edit alternative versions of the elaboration messages that are displayed after a student chooses an answer (all figures are screen shots from the experiment in Case Study 3). Instructors can then preview what a randomly chosen student will see (as illustrated in Figure 1), allowing them to reflect on potential experimental conditions in the context of students' experience. Figure 3 shows the dashboard presenting data about student ratings for each elaboration message, and the current probability of assigning students to these alternative conditions based on student ratings.

Data and Policy Dashboard

The data and policy dashboard enables instructors and researchers to view data in real-time, rather than waiting weeks or months for data munging and organization. Figure 3 shows the dashboard using a screen shot from Case Study 3 after all students had participated. Aggregating outcome variables

Version	Probability of Explanation	Mean Student Rating	Number of Students	Standard Deviation of Rating
1. Correct! This is where f' (the slope of f) is negative and thus where f is decreasing.	0.48	9.00	18.00	2.18
2. Correct! This is where f' (the slope of f) is negative and thus where f is decreasing. If you would like to review this sort of question further, you can ...	0.52	9.10	18.00	1.91

Figure 3. The Data and Policy Dashboard shows instructors the student ratings for each of the conditions in their experiment: Mean, Number of Students participating, and Standard Deviation. The policy is also shown – the probability of each condition being presented, at a given point in time. This dashboard shows what the instructor in Case Study 3 saw after all students had participated. The table in Figure 6 summarizes this dashboard information on the left, and on the right includes additional information for Case Study 3, e.g., Standard Error of the Mean, and the instructor's predictions about quality of elaboration messages.

by condition maintains student anonymity and privacy, while furnishing the instructor with the key data about the relative effectiveness of the conditions. The dashboard also shows the respective probabilities that each of the conditions will be assigned to the next student, to avoid the randomization policy being a "black box."

Dynamically Randomized Assignment

To enable instructors to turn data from experiments into practical improvements to problem content, DynamicProblem provides an algorithm for dynamically re-weighting the probability that a condition will be assigned to the next student, with the goal of providing future students with the highest rated elaboration messages. In brief, the probability of assigning a condition (elaboration message) is originally uniform random (e.g., 50/50), but becomes weighted random (e.g., 80/20) as data is collected. The probability of assigning a condition is proportional to how many past students have given that condition a higher rating than the alternatives.

Probability matching algorithm for dynamic experimentation

Deciding when there is enough data to be confident of choosing the best condition can be understood in terms of the *exploration* versus *exploitation* tradeoff explored in the reinforcement learning literature (see [32] for an overview). The algorithm for dynamically weighted randomization must balance exploiting information collected so far, such as continuing to select an explanation because the first student who saw it rated it highly, with exploring and collecting information about alternative explanations, which might be rated more highly by future students.

We formalize the decision about which conditions are assigned to students as a multi-armed bandit problem, a model used for automated online experimentation in websites [18], and more recently, for optimizing educational resources [33, 20]. In a multi-armed bandit problem, the system's goal is to maximize its total rewards. At each time point, it chooses one action from a set of possible actions, and receives a noisy reward based on the action chosen. In our application, rewards are student outcomes, and the learned *policy* specifies what action the system will take (which elaboration message will be provided).

A wide range of algorithms can be used for this problem. To aid in interpretability for instructors, we made the design choice to use a *probability matching* algorithm called Thompson Sampling [9] which can be understood as dynamically weighted randomization. The probability of assigning condition X to a student is equal to the current probability that condition X has the highest rating, based on the previously observed ratings, and an assumed statistical model. Our goal was that the system's assignment of conditions would not be a "black-box", but provide instructors a convenient interpretation of how and why conditions are assigned to students.

Our implementation used Bayesian inference over a Beta-Binomial model for the student ratings for each condition. We specifically used a Beta(19, 1) prior distribution, which can be understood as the prior belief that any condition has received one rating of a 9 and one rating of a 10. This represents optimistic but uncertain initial beliefs about each condition. Alternative prior beliefs could have been used without changing the overall system design. The Binomial likelihood function assumes the student rating of r follows a Binomial distribution with 10 samples, r of which are positive. The choice of the conjugate Beta-Binomial model allows efficient real-time updating of the policy, as the model's posterior distribution over each condition remains a Beta distribution.

Beta-Binomial (or Beta-Bernouili) models are popular in multi-armed bandit applications of Thompson Sampling, from optimizing websites to features of educational technology [30, 33, 20]. The novel approach in this paper is embedding it within an end-user tool for instructors, to provide an automated yet interpretable method for transitioning from data collection in a traditional experiment to providing useful conditions to future students.

REAL-WORLD CASE STUDIES OF DYNAMICPROBLEM

To understand and evaluate the instructor-centered approach instantiated in DynamicProblem, we present case study deployments into the courses of three instructors, who worked with the first author to conduct experiments. Our sample size was limited by the investment in full end-to-end design, deployment, and analysis of experiments, but conducting these case studies in real courses provide deeper insight into how the system would be used for realistic applications.

Participating Instructors

We identified three faculty at Harvard University who used the Canvas LMS to teach courses with a mathematics focus, where online problems and elaboration messages are widely used. These were a graduate-level methods course in a Public Policy school (102 students), an undergraduate course on Probability from the Statistics department (68 students), and an undergraduate course on Calculus (36 students). The instructors had never conducted experiments before with their students or any other population, but were familiar with the concept of randomized assignment as a scientific method. The instructors were invited to try out a new system for experimenting with elaboration messages.

Use of DynamicProblem for Experimentation

Either the researcher or instructor added the DynamicProblem system to each instructor's course in the Canvas LMS, using the functionality major platforms provide for including LTI tools. The researcher then met with each instructor to show them the DynamicProblem system and describe how it could be used to present different elaboration messages to students, and collect data about student ratings of helpfulness. The researcher asked the instructor for a description and examples of their current use of online problems, and what pedagogical questions about problem feedback that they might like to answer using DynamicProblem. Both the instructor and the researcher proposed multiple ideas for experiments testing feedback, and one was selected by consensus, anywhere from the first to third meeting. The operationalization of the experimental question and hypotheses into specific conditions (explanations, hints, learning tips) and implementation in DynamicProblem typically raised more discussion (over email or in person) and required 2-3 rounds of back and forth direct editing of elaboration messages. After final approval, the problem and embedded experiment was deployed to students.

Before the instructor viewed the data dashboard, the researcher suggested the instructor make quantifiable predictions about which conditions/feedback messages they thought would be more helpful for students' learning. Specifically, the instructor answered the prompt: "Rate how helpful you think each message will be for learning, on a scale from 0 (not at all helpful) to 10 (extremely helpful)". The instructor also judged "Rate how confident you are in your evaluation, on a scale from 1 (not at all confident) to 5 (perfectly confident)." Making predictions about experimental results in advance has been shown to reduce 'hindsight bias' or the sense that people knew the result all along [11]. These prompts aimed to get the instructor to reflect on their beliefs about which feedback messages would be effective, and how uncertain they were about these beliefs, which could better prepare them to understand how the data informed their beliefs and pedagogical practice. These ratings are shown in the right half of the table summarizing the experimental data for each case study.

1. Analogical Explanations in a Public Policy Course

Design. The instructor identified a problem to test a question they had been wondering about in previous offerings of the course – whether to introduce an analogy to telescopes in explaining the concept of the statistical power of a hypothesis test. The instructor compared two explanations: The first explained statistical power of a hypothesis test in terms of the distribution of a sampling statistic and critical regions. The second explanation provided an analogy comparing the power of a hypothesis test to detect true effects in a population, to the power of a telescope to detect objects: A smaller object (lower impact of a program) would require a more powerful telescope (a greater sample size to increase statistical power).

Results. The left half of Figure 4 summarizes the results of the experiment that appeared on the data dashboard, although instructors viewed this data using the interface in Figure 3. The right half of Figure 4 provides additional information that was not displayed in the dashboard, namely: The standard error

Version	Probability of Condition	Mean Student Rating	Number of Students	Standard Deviation of Rating	Standard Error of the Mean	Instructor Rating	Instructor Confidence
1. Quantitative Explanation	0.23	7.26	46.00	1.87	0.28	7/10	2/5
2. Analogical Explanation	0.77	7.48	56.00	1.59	0.21	5/10	2/5

Figure 4. Case Study 1 data: The left side of the table summarizes the information from the instructor dashboard after all students had finished, which the instructor and researcher would view using the interface shown in Figure 3. The key information was: The probability of assigning the next student to a condition according to the algorithm for dynamically weighted randomization, the current mean student rating, the number of students, and the standard deviation of the student ratings. The right side of the table shows the standard error of the mean (SEM) for student ratings, the instructor's rating of how helpful they predicted each message would be for student learning, and the instructor's confidence in their helpfulness rating.

Version	Probability of Condition	Mean Student Rating	Number of Students	Standard Deviation of Rating	Standard Error of the Mean	Instructor Rating	Instructor Confidence
1. Answer	0.00	2.50	2.00	2.50	1.77	6/10	2/5
2. Hint	0.00	5.83	6.00	2.67	1.09	6/10	2/5
3. Solution	1.00	7.88	60.00	2.47	0.32	9/10	2/5

Figure 5. Case Study 2 data: The left side shows the instructor dashboard after all students completed the problem. The right shows the standard error of the mean (SEM) for student ratings, and the instructor's rating (with confidence judgment) of how helpful they anticipated the messages would be.

of the mean for each condition, and the instructor's predictions (and confidence) about how effective each explanation would be. The data suggested a small trend for the analogical explanation to be rated higher, which surprised the instructor, who had predicted the opposite. Given the small effect, the instructor wondered if the trend would hold up with more data. The data did reduce the instructor's initial concerns that students would be frustrated or confused by the introduction of the analogical explanation, which was less technical and had not been covered in class.

2. Answers, Hints, Solutions in a Probability Course

Design. The instructor raised a tension that had arisen in previous teaching, about how much information to provide to students. The experiment therefore investigated how students would react to receiving just answers or hints, compared to full solutions. After students entered an answer, the instructor's experiment assigned them to receive either: (1) the correct answer (e.g. "Answer: 9 is more likely"); (2) a hint (e.g. "Hint: Dice are not indistinguishable particles!..."), or (3) A full worked-out solution to the problem (e.g. "Solution: Label the dice as Die 1 and Die 2 and label...").

Results. Figure 5 shows the results of the experiment as they appeared on the data dashboard, along with the instructor's predictions about which conditions would be best. The data suggested that students rated the full solution as most helpful, followed by the hint and then the answer alone. The instructor found the data to be a useful confirmation of their predictions, which they had wanted to verify. The instructor also commended the rapid roll-out of the higher rated solution: "Because you want to explore which option is better, but if one of them is clearly better, you want most students to be seeing that one."

3: Learning Tips in a Calculus Course

Design. The instructor was concerned that students solving problems only rarely made connections with or reviewed the concepts from previous lectures, which the instructor believed was valuable for learning. This motivated an experiment that compared: (1) the existing content-specific feedback message, like "Correct! This is the lowest point on the graph of f." to (2) the feedback message plus a general learning tip, like "If you would like to review this sort of question further, you can look back at your notes about how we found where the function g was greatest and least. Or you can look at the relevant video." The instructor wanted to examine both students' subjective ratings *and* a measure of learning. The experiment was therefore embedded in the first of a series of related problems on the graphs of derivatives.

Results. Figure 6 shows the information from the data and policy dashboard, as well as accuracy on the subsequent 4 problems. The instructor had predicted that students would

Version	Probability of Condition	Mean Student Rating	Number of Students	Standard Deviation of Rating	SEM - Rating	Next Problem Accuracy	SEM - Next Problem Accuracy	Instructor Rating	Instructor Confidence
1. Feedback	0.48	9.00	18.00	2.18	0.51	0.64	0.07	7/10	4/5
2. Feedback and Prompt to Review	0.52	9.10	18.00	1.91	0.45	0.72	0.07	8/10	4/5

Figure 6. Case Study 3 data: The left side shows the instructor dashboard after all students took the problem. The right side shows two variables unique to case study 3: "Next Problem Accuracy" is the mean accuracy in solving 4 related problems that students attempted, after doing the problem in which the experiment on messages was conducted; and the standard error of the mean for Next Problem Accuracy.

find the extra information far more helpful, and was surprised by students' ratings that the elaboration messages were equally helpful. This led the instructor to express the desire to run the experiment in future offerings to collect a larger sample of data, as well as try out other ways of framing the learning tip.

INSTRUCTORS' PERSPECTIVES ON EXPERIMENTS

This section presents observations from the case studies on the instructors' perspectives on conducting the experiments. These are based on field observations collected while the first author worked with the instructor in using DynamicProblem. In addition, each instructor participated in a follow-up interview after the conclusion of the research, to share their experiences. All these meetings were recorded and transcribed. These transcripts were reviewed by the first author to identify themes that pertained to the challenges and design goals introduced earlier in the paper, concerning how instructors' experience in conducting experiments related to their instructional goals.

Instructors were empowered to conduct experiments

DynamicProblem was designed to understand how instructors might use experiments to improve their pedagogy. Discussion with the instructors indicated that they had not previously used experiments as a tool in their own classes. But this effort led them to see experiments as *"a valuable tool...[for] the teacher to understand how their students learn. Not just in broad terms but specifically in their course."* (I2: Instructor 2).

All three case studies resulted in instructors successfully co-designing experiments with a researcher, and the experiments were successfully deployed within their classes, demonstrating that DynamicProblem did facilitate experimentation. To design and deploy an experiment required instructors to spend approximately 2-6 hours in total, and the researcher to spend 3-5 hours on each experiment. This is a small amount of time compared to experiments run on math problems by the first author in the past. These could take as much as 20-40 hours because of back-and-forth in designing experimental conditions, figuring out how to embed a randomized experiment in online platform and then writing code, and determining where data was collected and how it could be linked to randomization.

Experimentation led to reflection on instructional design

One hope was that DynamicProblem would make experiments better meet instructors' goals, such as coming up with new ideas and reflecting on how they designed elaboration messages. I3 noted that previously they were just focused on "typing up one version" to get the problem complete, and the scaffolding that they provided tended to be *"very specific to the question at hand."* However, designing experiments led I3 to more actively focus on the implications of different instructional choices, and be more creative about general strategies for supporting students: *"adding a link back to a video or a link back to a chalkboard is something I had not previously considered."* This suggests that thinking through the design of experiments may be helpful for instructors to improve their teaching: I3 said *"So even if we don't get any significant data, that will have been a benefit in my mind."*.

Data from experiments was seen as informing pedagogy

In addition to helping instructors think about their pedagogy more broadly, the data collected in DynamicProblems and visualized on the data dashboard was helpful for instructors to better understand their students' learning. For example, I2 noted that they usually found the data analytics in their learning management system *"not very useful"* and too focused on things like the timing of when students completed a problem. Because DynamicProblem asked students to rate the helpfulness of explanations, the instructors could gain a better understanding of students' experiences in the problem from the dashboard, as this information was aggregated on the screen. I3 mentioned appreciating seeing this broken down by experimental condition, which provided *"data that I normally don't get"* about the helpfulness of scaffolding. The availability of data from experiments helped instructors to reflect on alternative design decisions: The data from Case Study 1 was inconsistent with I1's prior beliefs, as they predicted mathematical explanations would be rated by students as more helpful than analogical explanations.

Dynamic Experimentation as a practical tool for improving a course

Instructors saw the practical value of dynamic experiments for improving curricular materials and helping students, not

just for the sake of research. I3 said *"I think it is best for the students, it adds value that students are more likely to get the explanation that's going to help them"* and I2 echoed that it *"improved the experience of many of the students by giving them answers that are more helpful... the students updating earlier are helping the ones who did it later.... That's pretty neat."*

Instructors were more willing to conduct experiments when they knew the technology could rapidly deploy better conditions into the actual problem, and I1 planned to continue running the experiment *"from one semester to another, that is going to increasingly refine which versions are the best and feed those to the students."* In contrast, many educational field experiments lead to reports or papers about results, but there are many delays in using data from the experiment to change the course content or experience of the students.

Instructors were more motivated and informed about conducting future experiments

The simplicity of the initial experiments allowed instructors to deploy experiments after hours instead of weeks. This could raise a concern that such tools might simply lower the barriers to having instructors run low quality experiments. For example, learning researchers might be concerned that the outcome variables in these experiments are students' ratings, not measures of learning. On the other hand, grappling with the challenges researchers face in designing experiments could make instructors more aware of the need for such research. We found that in the course of conducting their first experiment, instructors began reflecting on better ways to measure the effectiveness of elaboration messages. For example, in identifying follow-up 'post-test" problems that could be included to measures how receiving an elaboration message influenced learning. I3 did in fact include related problems as a measure of learning, as shown in Figure 6.

I2 wanted to build on their comparison of hints to solutions and compare *"multiple full solutions,"*, intending to compare *"one approach that's pretty long and tedious but very easy to understand and another one that's much more elegant, but harder to understand why it works."* I3 saw particular value for experiments *"when we add new content to the course that I haven't had as much experience teaching."* The benefits of DynamicProblem are therefore not limited to a single experiment, but helping instructors enhance their instruction by coming up with new ideas and repeatedly experimenting.

DISCUSSION, LIMITATIONS & FUTURE WORK

Instructors in our three case studies reported that designing experiments about practical questions of interest helped them reflect on their pedagogy, and that the system's immediate access to data and use of machine learning for dynamic experimentation provided a convenient pathway to enhancing problems for future students.

We hope this is only the first of many papers to investigate instructor-centered experimentation. Digital experimentation could become a valuable tool for instructors to collect data about how instructional design impacts students, but there are many questions for future research to answer about the relationship between randomized experiments and instructional practice. Instructor-centered experimentation may also facilitate closer collaboration between instructors and researchers, ensuring that interventions can be adapted to real classrooms and that experiments address practical problems and are attentive to ethical issues.

To help provide a foundation for this future work, we consider specific limitations of how we evaluated this approach, and then turn to current limitations of an instructor centered approach more broadly, and what questions these limitations raise for future research.

This paper involved instructors in end-to-end field deployments, in order to deeply explore a small number of case studies. But this could limit the generalizability of the insights. Future work needs to explore a greater number and variety of instructors. Beyond mathematics problems, instructors in other disciplines provide online problems and activities that could benefit from testing elaboration messages. The approach we present also should be tested for a broader range of experiment types, such as experimenting with components of online courses other than individual problems.

Since the first author worked with instructors in the case studies, we were concerned the system might be difficult to use by instructional designers with less support. We conducted additional informal tests of usability. We provided DynamicProblem to a class of 20 master's students in education as a course exercise in testing out features of a problem, along with written instructions about the system functionality. Working in six groups of 2-3 students, all were able to build and test out on each other a problem containing an experiment with feedback, in 1-1.5 hours. We also showed the tool to four teaching assistants and asked them to use DynamicProblem to replicate the experiment from Case Study 2, with no assistance or documentation. All were able to complete the task. They remarked that it was *"convenient to author the explanations in the same interface as the problems"* and they would not need assistance in implementing as *"one person could do it."*

One limitation of DynamicProblem is that it collects students' subjective ratings of quiz content rather than measuring learning. Educational researchers often carefully design assessments of knowledge and learning, as people's self-reports about how well they understand a concept can be unreliable [29]. It should be noted that we did not ask students how well they understand a general topic, but asked them to rate how helpful an elaboration message was for learning from a specific problem, which has been shown to be predictive of subsequent learning from explanations [33]. Our aim with ratings of elaboration messages was to rapidly provide instructors with an analog of information they perceive and use when teaching in person – students' expression of confusion or understanding. Future systems optimize for behavioral measures of knowledge, such as accuracy on future problems.

To understand student perspectives on being in experiments, we administered an optional survey in Case Study 3. The survey explained that students were randomly assigned to alternative versions of the feedback messages and ratings were collected in order to figure out which were most helpful. Stu-

dents were asked to quantify their attitude towards being randomly assigned, and the mean response was 2.52 on a scale from -5 (strongly disapprove) to +5 (strongly approve). Students commented that *"I've been involved with usability and A/B testing on web site development, so it doesn't shock me"*, *"I like the idea"*, and that it *"Fine tunes the learning for future students"*. Students did not express serious objections to the experiment, and in fact appreciated that it would improve the course and help fellow students.

More broadly, there are limitations to the instructor-centered approach we propose. There are of course many instructional design questions that are not best answered through randomized experiments, but through alternative approaches, like the qualitative analysis and design methods used in action research [24]. Class sizes are a limitation on the value of experimental data, making large classes, or those with multiple parallel offerings, most appropriate.

Future work should also explore how and whether instructor-centered approaches can simultaneously meet researchers' needs. Systems like DynamicProblem may be helpful to researchers by making it easier to embed experiments in courses, facilitating effective collaboration with instructors. However, future work should investigate researchers' openness and ability to analyze dynamically re-weighted experiments (as is being done in other contexts, e.g., [30, 5]). Additional structure may be needed for instructors and researchers to identify experiments that are of mutual scientific and practical interest. One possibility is that researchers and instructors can meet their differing goals by running different experiments, or including different conditions in the same experiment. This raises the question of when and how instructors will be able to run experiments that are useful and rigorous enough to improve their instruction and help students.

While a successful experiment for a researcher is one that contributes to scientific knowledge, a successful experiment for an instructor may be one that enhances their typical instructional practice. None of the instructors in our case studies had previously conducted laboratory or field experiments, but they reported that designing the experiments helped them reflect on pedagogical decisions and generate new ideas, even before any data was available.

A related limitation is whether instructors need extensive statistical knowledge in order to interpret data from experiments and use it to make decisions. The dynamic experimentation algorithm does mean that instructors don't have to make moment by moment decisions about whether to stop an experiment, as past work has provided near-optimal bounds on how the regret (i.e., the difference in outcomes between the chosen condition and the best condition) of the algorithm's choices grows over time [1]. But future work can explore whether providing statistical training or changing the kind and format of data presentation results in better use of experimental results by instructors, or if there are benefits to allowing instructors greater control over the algorithm.

One additional limitation of our approach is the lack of attention to discovering which conditions work for different subgroups of students, such as whether quantitative or analogical explanations were more helpful for students with low versus high prior knowledge. In fairness, most experiments first focus on the difficult task of discovering "what works", as a precusor to "what works for which type of student?". Such an extension to instructor-centered experimentation may require significant future investigation of instructors' needs and ability to collect data about individual differences and analyze data for complex statistical interactions. One promising technological direction could be *contextual* bandit algorithms for dynamic assignment that can take into account contextual information about learners themselves (e.g., [19]). Such work would build on our current implementation of algorithms that turn randomized experiments into engines for dynamic improvement, by turning tools for experimentation into engines for dynamic personalization.

CONCLUSION

To help realize the promise of experimentation to help instructors enhance digital educational resources, this paper presented design guidelines for more instructor-centered tools for experimentation. Our goal was to reduce the programming knowledge and time that instructors need to conduct experiments, and to help instructors more rapidly use data from each student to enhance the learning experience of the next student. We instantiated a solution in the proof-of-concept system DynamicProblem, which lowered the barriers conduct and obtain data from experiments on elaboration messages in online problems, such as explanations, hints, and learning tips. The system used a multi-armed bandit algorithm to analyze and use data about students' ratings to present the higher rated conditions to future students, automatically enhancing the problems. Case study deployments with three instructors suggested the system helped them reflect on how to improve pedagogy, and provided a data-driven pathway for enhancing their online problems. Just as decades of work have established how learning researchers can effectively use experiments, a great deal of future work is needed to understand how instructors can effectively use experiments. We hope that this paper provides a foundation for these future investigations of instructor-centered experimentation.

ACKNOWLEDGEMENTS

This work was supported in part by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korean government (MSIT) (No.20160005640022007, Development of Intelligent Interaction Technology Based on Context Awareness and Human Intention Understanding), as well as by the UM Poverty Solutions program and the IBM Sapphire Project at the University of Michigan.

REFERENCES

1. Shipa Agrawal and Navin Goyal. 2013. Further Optimal Regret Bounds for Thompson Sampling. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*. Journal of Machine Learning Research, 99–107.
2. T. Anderson and J. Shattuck. 2012. Design-Based Research: A Decade of Progress in Education Research?

41, 1 (2012), 16–25. DOI: http://dx.doi.org/10.3102/0013189X11428813

3. Sasha Barab and Kurt Squire. 2004. Design-based research: Putting a stake in the ground. *The journal of the learning sciences* 13, 1 (2004), 1–14.
4. Lecia Barker, Christopher Lynnly Hovey, and Jane Gruning. 2015. What Influences CS Faculty to Adopt Teaching Practices?. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15)*. ACM, New York, NY, USA, 604–609. DOI:http://dx.doi.org/10.1145/2676723.2677282
5. Scott M Berry, Bradley P Carlin, J Jack Lee, and Peter Muller. 2010. *Bayesian adaptive methods for clinical trials*. CRC press.
6. Kirsten Berthold, Matthias Nückles, and Alexander Renkl. 2007. Do learning protocols support learning strategies and outcomes? The role of cognitive and metacognitive prompts. *Learning and Instruction* 17, 5 (2007), 564–577.
7. Ann L Brown. 1992. Design experiments: Theoretical and methodological challenges in creating complex interventions in classroom settings. *The journal of the learning sciences* 2, 2 (1992), 141–178.
8. Anthony S Bryk, Louis M Gomez, and Alicia Grunow. 2011. Getting ideas into action: Building networked improvement communities in education. In *Frontiers in sociology of education*. Springer, 127–162.
9. Olivier Chapelle and Lihong Li. 2011. An empirical evaluation of thompson sampling. In *Advances in neural information processing systems*. 2249–2257.
10. Mingyu Feng, Neil Heffernan, and Kenneth Koedinger. 2009. Addressing the assessment challenge with an online system that tutors as it assesses. *User Modeling and User-Adapted Interaction* 19, 3 (2009), 243–266.
11. Scott A Hawkins and Reid Hastie. 1990. Hindsight: Biased judgments of past events after the outcomes are known. *Psychological Bulletin* 107, 3 (1990), 311.
12. Gillian R. hayes. 2011. The Relationship of Action Research to Human-computer Interaction. *ACM Trans. Comput.-Hum. Interact.* 18, 3, Article 15 (Aug. 2011), 20 pages. DOI:http://dx.doi.org/10.1145/1993060.1993065
13. Neil T Heffernan and Cristina Lindquist Heffernan. 2014. The ASSISTments Ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education* 24, 4 (2014), 470–497.
14. Benjamin Mako Hill, Dharma Dailey, Richard T Guy, Ben Lewis, Mika Matsuzaki, and Jonathan T Morgan. 2017. Democratizing Data Science: The Community Data Science Workshops and Classes. In *Big Data Factories*. Springer, 115–135.
15. Samad Kardan and Cristina Conati. 2015. Providing Adaptive Support in an Interactive Simulation for Learning: An Experimental Evaluation. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 3671–3680. DOI: http://dx.doi.org/10.1145/2702123.2702424
16. Kenneth R Koedinger, Jihee Kim, Julianna Zhuxin Jia, Elizabeth A McLaughlin, and Norman L Bier. 2015. Learning is not a spectator sport: Doing is better than watching for learning from a MOOC. In *Proceedings of the Second (2015) ACM Conference on Learning@ Scale*. ACM, 111–120.
17. Ron Kohavi, Roger Longbotham, Dan Sommerfield, and Randal M Henne. 2009. Controlled experiments on the web: survey and practical guide. *Data mining and knowledge discovery* 18, 1 (2009), 140–181.
18. John Langford and Tong Zhang. 2008. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in neural information processing systems*. 817–824.
19. Lihong Li, Wei Chu, John Langford, and Robert E Schapire. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*. ACM, 661–670.
20. J Derek Lomas, Jodi Forlizzi, Nikhil Poonwala, Nirmal Patel, Sharan Shodhan, Kishan Patel, Ken Koedinger, and Emma Brunskill. 2016. Interface Design Optimization as a Multi-Armed Bandit Problem. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 4142–4153.
21. Marsha Lovett, Oded Meyer, and Candace Thille. 2008. JIME-The open learning initiative: Measuring the effectiveness of the OLI statistics course in accelerating student learning. *Journal of Interactive Media in Education* 2008, 1 (2008).
22. J Nathan Matias. 2016. Going Dark: Social Factors in Collective Action Against Platform Operators in the Reddit Blackout. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1138–1151.
23. G Mcfall and L Neumann. 2010. IMS learning tools interoperability basic LTI implementation guide v1. 0 Final. *Internet: http://www. imsglobal. org/lti/2010 [May 2013]* (2010).
24. James E McLean. 1995. *Improving Education through Action Research: A Guide for Administrators and Teachers. The Practicing Administrator's Leadership Series. Roadmaps to Success.* Corwin Press.
25. Korinn S Ostrow, Doug Selent, Yan Wang, Eric G Van Inwegen, Neil T Heffernan, and Joseph Jay Williams. 2016. The assessment of learning infrastructure (ALI): the theory, practice, and scalability of automated assessment. In *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge*. ACM, 279–288.

26. William R. Penuel, Anna-Ruth Allen, Cynthia E. Coburn, and Caitlin Farrell. 2015. Conceptualizing Research-Practice Partnerships as Joint Work at Boundaries. *Journal of Education for Students Placed at Risk (JESPAR)* 20, 1-2 (2015), 182–197. DOI: http://dx.doi.org/10.1080/10824669.2014.988334

27. Leena Razzaq and Neil Heffernan. 2006. Scaffolding vs. hints in the Assistment System. In *Intelligent Tutoring Systems*. Springer Berlin/Heidelberg, 635–644.

28. Alexander Renkl. 1997. Learning from worked-out examples: A study on individual differences. *Cognitive science* 21, 1 (1997), 1–29.

29. Leonid Rozenblit and Frank Keil. 2002. The misunderstood limits of folk science: An illusion of explanatory depth. *Cognitive Science* 26, 5 (2002), 521–562.

30. Steven L Scott. 2010. A modern Bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry* 26, 6 (2010), 639–658.

31. Valerie J Shute. 2008. Focus on formative feedback. *Review of educational research* 78, 1 (2008), 153–189.

32. Richard S Sutton and Andrew G Barto. 1998. *Reinforcement learning: An introduction*. Vol. 1. MIT press Cambridge.

33. Joseph Jay Williams, Juho Kim, Anna Rafferty, Samuel Maldonado, Krzysztof Z Gajos, Walter S Lasecki, and Neil Heffernan. 2016. AXIS: Generating Explanations at Scale with Learnersourcing and Machine Learning. In *Proceedings of the Third (2016) ACM Conference on Learning at Scale*. ACM, 379–388.

34. Joseph Jay Williams, René F Kizilcec, Daniel M Russell, and Scott R Klemmer. 2014. Learning innovation at scale. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems*. ACM, 131–134.

35. Joerg Wittwer and Alexander Renkl. 2008. Why instructional explanations often do not work: A framework for understanding the effectiveness of instructional explanations. *Educational Psychologist* 43, 1 (2008), 49–64.

36. Hsiu-Ping Yueh, Tzy-Ling Chen, Weijane Lin, and Horn-Jiunn Sheen. 2014. Developing digital courseware for a virtual nano-biotechnology laboratory: A design-based research approach. *Journal of Educational Technology & Society* 17, 2 (2014).