The Effects of Spaced Repetition in Online Education

By Dillon Dumesnil

S.B., Computer Science and Engineering, MIT, 2017

Submitted to the

Department of Electrical Engineering and Computer Science

In Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

Massachusetts Institute of Technology

June, 2018

Author: ____

Department of Electrical Engineering and Computer Science
May 25, 2018

Certified by: ____

Sanjay Sarma, Vice President for Open Learning, Thesis Supervisor
May 25, 2018

Certified by: ____

Mike Dikan, Software Engineering Manager, edX, Thesis Co-Supervisor
May 25, 2018

Accepted by: ____

Katrina LaCurts, Chair, Master of Engineering Thesis Committee

The Effects of Spaced Repetition in Online Education

By Dillon Dumesnil

Submitted to the
Department of Electrical Engineering and Computer Science
on May 25, 2018
In Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

Over the course of the past year, I have designed and implemented a tool to simulate spaced repetition in Massively Open Online Courses hosted on edX. I created the tool using edX's course building block, the XBlock, and then ran an experiment in three courses to evaluate its effectiveness. My research was set up as an A/B experiment with a 50% split between the experimental and control groups. After running in two of the courses for approximately one month each and the third course for six months, 32 unique users interacted with the tool. Out of the learners who used the tool, there was a slight increase in their average grade compared to those that did not use it.

Thesis Supervisor: Sanjay Sarma, Vice President for Open Learning
Thesis Co-Supervisor: Mike Dikan, Software Engineering Manager, edX

Acknowledgments

I would like to first express my sincere gratitude to Mike Dikan for his technical and writing advice, Professor Sanjay Sarma for coming up with a project idea and advising my thesis writing, and Professor Isaac Chuang for his assistance in data analysis and asking the right questions. I would also like to thank Megan Frank for making my co-op possible and all of the people at edX who helped advise, design, and review my code for the Review XBlock. A final thank you to my family who supported me throughout my entire academic career and my friends who helped keep me sane.

Table of Contents

1. Introduction **13**

2. Related Work **17**

3. Implementation **20**

3.1. EdX Course Architecture 20

3.2. Review XBlock 24

3.2.1. Motivation 24

3.2.2. Design 26

3.3. Shadow Course 37

4. Experiment Setup **40**

5. Experiment Analysis and Results **42**

5.1 Descriptive Statistics 43

5.2 Multiple Regression Analysis 49

6. Improvements **53**

7. Conclusions **55**

Appendix A **58**

List of Figures

Figure 1. Ebbinghaus Forgetting Curve 16

Figure 2. Leitner System 18

Figure 3. Example of Chapters and Sequentials in an edX course 22

Figure 4. Example of Verticals in an edX course 23

Figure 5. EdX LTI Consumer Example 29

Figure 6. Review XBlock 31

Figure 7. Review XBlock Expanded 35

Figure 8. Control Group View 40

List of Tables

Table 1. Subpopulations from Quantum Information Science I, Part 1 43

Table 2. Average Course Grades across Experiment Groups 44

Table 3. Average Course Grades across Experiment Groups and Submission of Review Problem 45

Table 4. Average Number of Events across Experiment Groups 46

Table 5. Average Number of Events across Experiment Groups and Submission of Review Problem 46

Table 6. Multiple Regression 1 evaluating Course Grade 50

Table 7. Multiple Regression 2 evaluating Course Grade 50

Table 8. Multiple Regression 3 evaluating Course Grade 51

Table A-1. Average Percentage of Users who earned a Certificate across Experiment Groups and Submission of Review Problem 58

Table A-2. Average Percentage of Users who hold a Bachelors or higher across Experiment Groups and Submission of Review Problem 58

Table A-3. Average Percentage of Users that are from the USA across Experiment Groups and Submission of Review Problem 59

Table A-4. Average Percentage of Users that are Female (used as a binary indicator for gender) across Experiment Groups and Submission of Review Problem 59

Table A-5. Average Age across Experiment Groups and Submission of Review Problem 60

1. Introduction

The beginning to massive online education began in 1999 with its inception at the Massachusetts Institute of Technology (MIT). It was in this year that the concept of MIT's OpenCourseware (OCW) was first designed and in 2002, OCW was launched with material from fifty MIT courses (Abelson 2008). The goal of OCW was "to publish the materials from all MIT undergraduate and graduate subjects freely and openly on the Web for permanent worldwide use" (Abelson 2008). By November of 2006, MIT's OCW featured 1600 courses and became a worldwide learning tool. The success of OCW set the stage for new educational platforms to be created that featured more interactive content.

Massive Open Online Courses (MOOCs) were first introduced to the world in the mid 2000's, but really began to grow in the early 2010's. In 2006, Khan Academy was founded through the video postings of Sal Khan and became the first major provider of online educational content at the high school level. Then in 2011 and 2012, three of the largest MOOC providers, edX, Coursera, and Udacity, were founded. By 2014, edX, Coursera, and Udacity had over ten million learners between them (Taneja, Shilpi, and Goel 2014).

Similar to Khan, Sebastian Thrun, a Stanford professor, began Udacity in 2011 by personally creating videos and posting them (Chafkin 2013). Thrun originally started Udacity as a way of creating free online education. He explains some of his reasoning behind creating online content, "I could restrict myself to helping a class of 20 insanely smart Stanford students who would be fine without me. But how could that impact not be dwarfed by teaching 160,000 students?" (Clafkin 2013). Despite being a major component towards the creation of online education, Thrun became critical of MOOCs after a couple of years working with Udacity. Thrun was frustrated with the low completion rates in MOOCs (approximately ten percent on average), and worked hard to improve them to no avail (Clafkin 2013). To this day, completion rates still remain low, and even out of those that complete the course, not all of them pass.

MOOCs tend to feature a course outline that takes the shape similar to regular courses at the high school and college levels. Learners who enroll in a course are taken to a series of lectures and assessments. The lectures are usually composed of several short lecture videos each explaining a small component of the current lesson. The style of short lecture videos originally came from Khan Academy's version of material. Accompanying the videos in a lesson are often brief text explanations about the material and potentially even brief in-lecture assessments so learners can gauge their level of understanding.

As the different companies progressed, it became more important to improve the educational experience for learners. This has been done in numerous ways. Many started offering certificates (for a price) to verify completion in a course with a particular grade percentage in the course. These certificates could then be used as proof of competency in a subject or topic and potentially benefit the learner in a working environment. As many MOOC providers are affiliated with various universities who provide the course content, the hope is that one day certificates can be used as substitute course credit. In 2012, Anant Agarwal, Founder and CEO of edX, was already looking forward to this idea, “He expects students will one day arrive on campus with MOOC credits the way they do now with Advanced Placement” (Pappano 2012). Working towards that end, edX launched a program in 2016 with the Massachusetts Institute of Technology (MIT) where learners could sign up for a series of related courses in a program called a MicroMasters. Upon completion of all of the courses with a high enough grade and in possession of the Verified Certificates, learners accepted to MIT’s Supply Chain Management Master’s Program would be considered to have the first semester of work done.

Another way online education can be improved is by offering a more personalized experience for learners. This thesis project has taken another step towards that goal by incorporating spaced repetition into a handful of edX courses. Spaced repetition is a learning technique that has been around for over a century, but never truly gained widespread traction in a traditional education setting. The technique incorporates increasing intervals of time between subsequent review of previously learned material, and can have positive effects on retention of information and overall understanding of material. Figure 1 contains a graphical representation on what this can mean for learners.

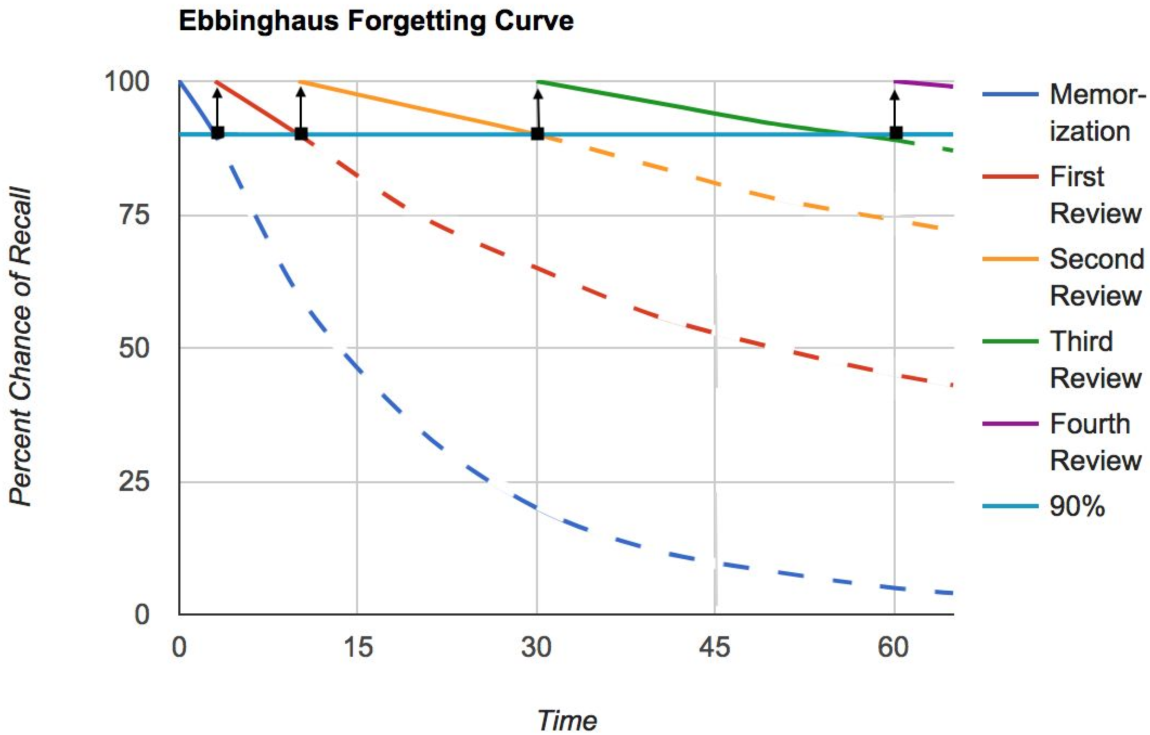

Figure 1. Ebbinghaus Forgetting Curve.
This image depicts an Ebbinghaus forgetting curve, first described in 1885 in *Über das Gedächtnis* (later translated to English as *Memory. A Contribution to Experimental Psychology*) (Ebbinghaus 2013). It also demonstrates the effects spaced repetition can have by showing a projected forgetting curve that becomes less steep with each reminder the learner receives indicating the learner has a better chance of remembering for longer periods of time after each review.

2. Related Work

The first reference to the idea of spaced repetition came from a German psychologist, Hermann Ebbinghaus. In 1885, Ebbinghaus introduced the ideas of a learning curve and a forgetting curve for human memory (Ebbinghaus 2013). In his research, Ebbinghaus experimented on himself to determine the effects of forgetting and came up with the idea that humans begin to forget at an exponential rate from original memorization. In addition to reporting about the rate of forgetting, Ebbinghaus found that as people review, the forgetting curve becomes less steep and as a result, need to review less often (see Figure 1). He called this concept the “spacing effect” and it remains the foundation for all future work in spaced learning (Ebbinghaus 2013). In 2015, Murre and Dros ran a study to test the validity of the Ebbinghaus forgetting curve. The experiment was designed similar to how Ebbinghaus originally did his with one of the authors (Dros) trying to memorize lists of nonsense syllables. Through their test, Murre and Dros confirmed the forgetting curve as originally designed remains consistent even in today’s world (Murre and Dros 2015).

Despite Ebbinghaus’ late 19th century work, modern spaced repetition had its modern roots in the 1970s and 80s with Sebastian Leitner and Piotr Wozniak. Leitner created what is known as the Leitner System in the early 1970s, an implementation of spaced repetition that featured a series of five boxes containing flashcards for review. Box one was reviewed on a daily basis, box two after a longer period of time (every other day), box three might only be reviewed once a week, and so on where the last box was rarely reviewed. New information to review would be placed in the first box and reviewed that day. If the information was correctly remembered, the flashcard would move back a box and so be reviewed less often. However, if the information was not remembered, it would go immediately back to the first box and back through the progression of boxes as shown in Figure 2. If a card reaches the final box and the information is correctly remembered, it is removed from the boxes with the implication that it is now in permanent memory. There is also a modified version of the Leitner System in which incorrectly answered flashcards only move back a single box rather than all the way to box one after a single mistake.

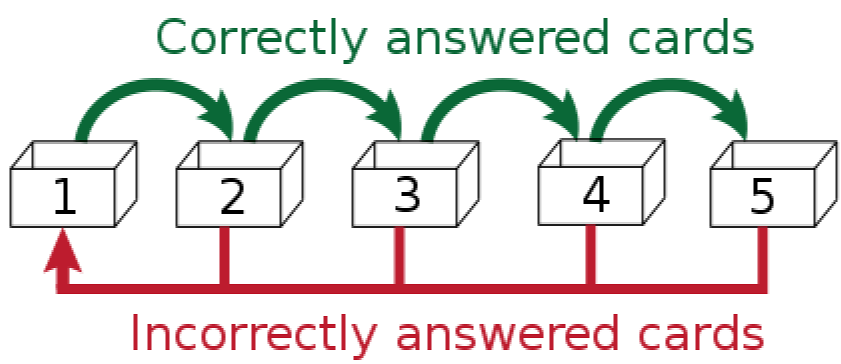

Figure 2. Zirguezi. Leitner System. Digital image. Leitner System. Wikipedia, 19 July 2012. Web. 22 June 2017.
The Leitner System was a simple implementation of spaced repetition. Flashcards would move to a later box when correctly answered for less frequent review and would be moved to the first box when not remembered for more frequent study.

Where Leitner created a simple implementation of spaced repetition, it was Wozniak who made it broadly applicable and usable. Over the course of the late 1980s and early 1990s, Wozniak created SuperMemo, a software package that implements spaced repetition from user-created content. The algorithm SuperMemo uses is similar to the Leitner System because as a user remembers a flashcard, the flashcard is presented to the user less often. It also offers a number of additional features including the user giving a response to how easily remembered the information was and being able to judge when to review again based on the difficulty a user had. Wozniak co-founded SuperMemo World in 1991 to distribute the SuperMemo algorithm.

Where SuperMemo and the Leitner system were focused on repeating flashcards, my project is expanding on this idea to repeat key concepts. Learners going through the experiment will potentially run into a handful of problems all dealing with the same material. The importance of this distinction is the change from pure information memorization to actually reviewing related content through learners' ability to solve problems and answer conceptual questions.

Other experiments involving spaced learning have been conducted over the years. One such experiment was done using US urology residents split into two groups. The study lasted for sixteen weeks and long-term retention was evaluated during weeks eighteen to forty-five. The results found that long-term knowledge was increased by 15.2% (Kerfoot, B. Price, et al. 2010).

3. Implementation

3.1. EdX Course Architecture

To help explain how I implemented my spaced repetition tool inside of edX, it is helpful to have an understanding of the edX course architecture. EdX courses are created in a tree structure, where each node has a single parent and can have multiple children. The higher levels of a course are fairly consistent across all courses and are composed of common features, such as sections and sub-sections. Only the leaves of the tree actually display content, while the other levels form the structure of the course. All nodes of the course tree are an edX datatype called an XBlock, which is the basic building block of edX. The ability of XBlocks to build on top of each other and form parent — children relationships is crucial to edX's infrastructure and many of their functions.

To go into the specifics of the structure, the top-level node is the *Course* block. The Course block essentially acts as the root node of the tree and thus a wrapper around the rest of the course. The direct descendants of the Course block are called *Chapter* blocks. The easiest way to think about Chapter blocks are as the sections of a course. A common way to use Chapters are as weekly breaks in the course. For example, a twelve week course would contain twelve Chapter blocks with each one labelled with the corresponding week.

The children of Chapter blocks are called *Sequentials* in the edX framework. Sequentials can be thought of as subsections and are essentially collections of content. Although the Sequential itself does not contain any content, it is designed to contain a specific type of of course material. An example of a Sequential would be a subsection for one lecture in a week or a problem set assigned to a particular week. It can also contain other miscellaneous information for a week, such as additional review information for a week or surveys that are typically conducted in the first and last weeks of a course.

The final generic type of XBlock is the *Vertical* block, the child of a Sequential. Keeping with the trend of the earlier definitions, Verticals are able to most easily be defined as units within a subsection. To briefly recap, the three main elements covered so far are Chapters (sections), Sequentials (subsections), and Verticals (units). Verticals, similar to Sequentials, can contain a collection of XBlocks inside of them. The XBlocks inside of Verticals will end up being displayed to learners as educational content and are the leaves of the defined course tree. Figures 3 and 4 show what all of these different XBlocks look like from a learners view of the course page.

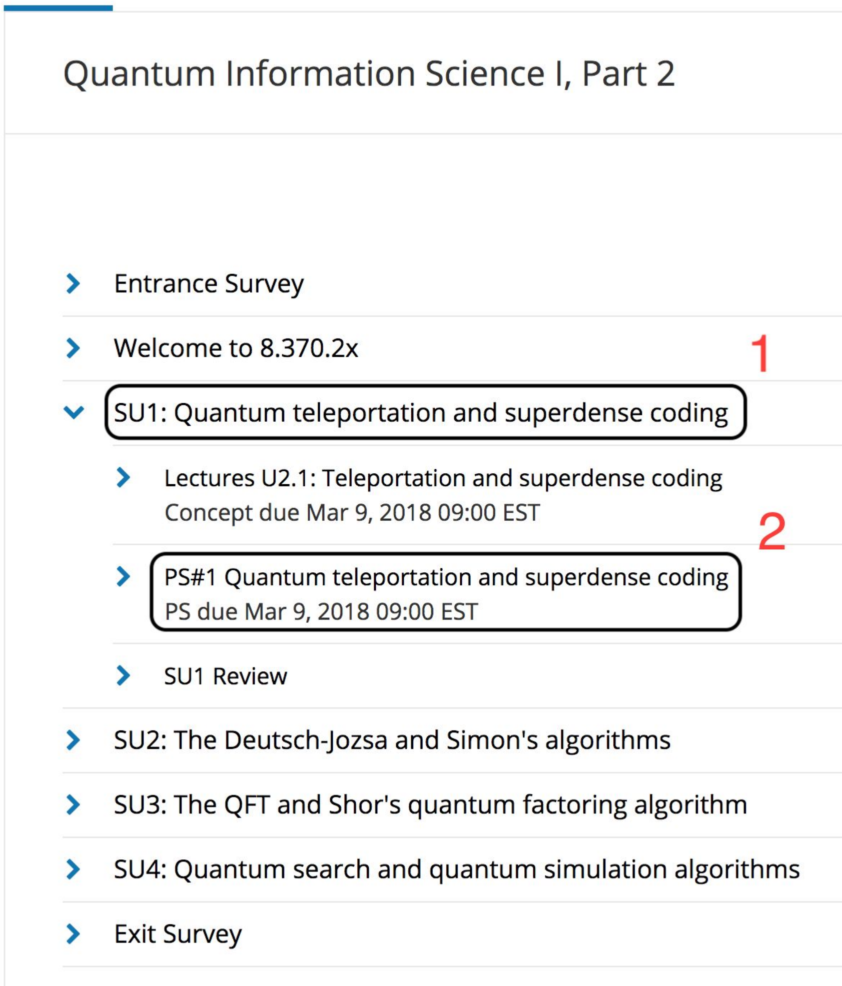

Figure 3. Screenshot from "MITx: 8.370.2x Quantum Information Science I, Part 2." — Example of Chapters and Sequentials in an edX course.

The box next numbered 1 is an example of a *Chapter*, or section. Above and below the boxed section, there are several other sections that exist inside of the course for the different weeks of the course. The box numbered 2 is a single subsection, or *Sequential*. Again, it is easy to see how each Sequential makes up a cohesive part of the week in this course.

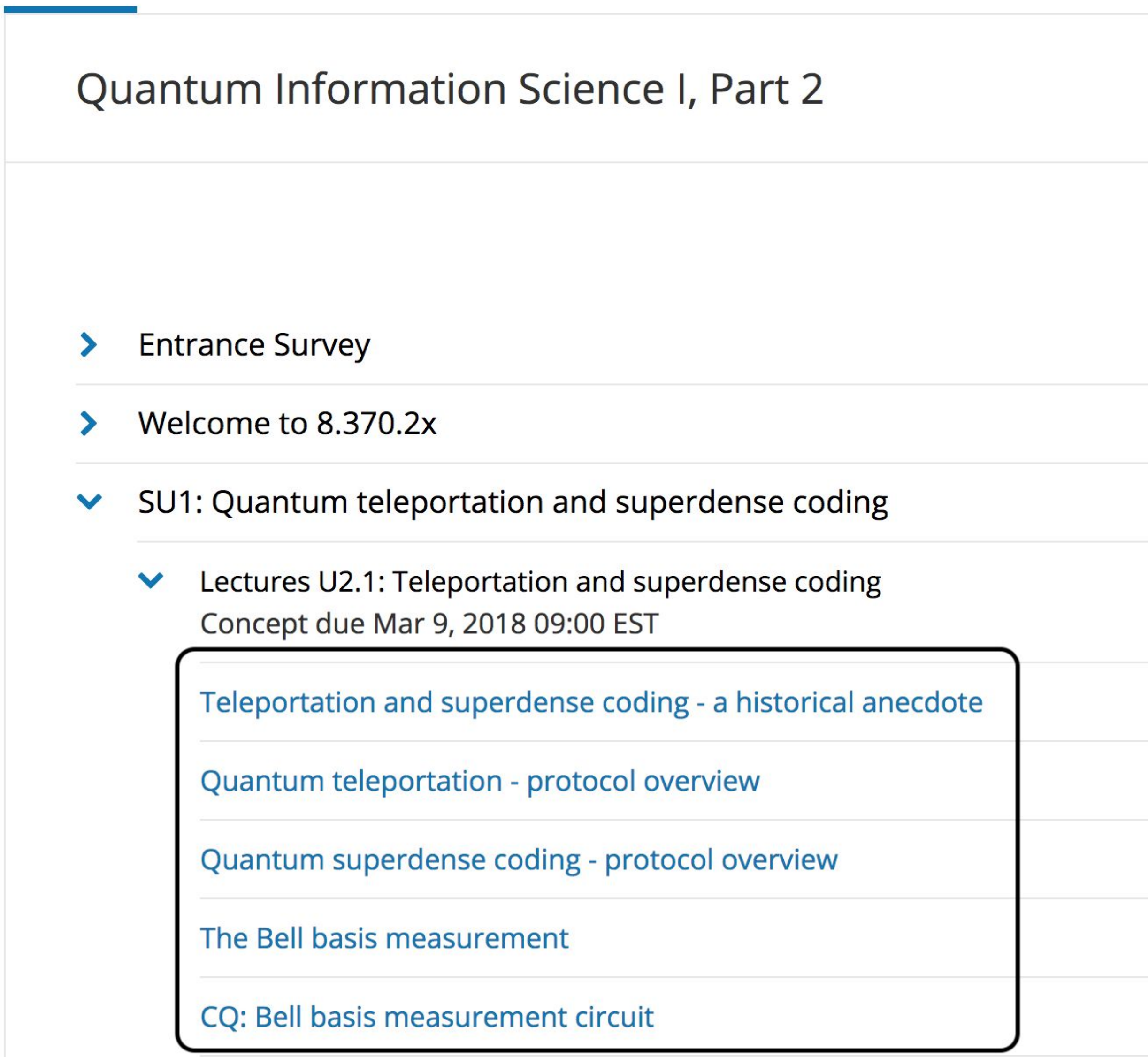

Figure 4. Screenshot from "MITx: 8.370.2x Quantum Information Science I, Part 2." — Example of Verticals in an edX course.

The boxed content in the image are all examples of *Verticals*, or units. As can be seen in the image, the units exist inside of a section to subsection to unit hierarchy. This particular set of units also shows a common practice among course instructors. The unit titled, “The Bell basis measurement,” is a video lecture and the unit directly following it, “CQ: Bell Basis measurement circuit,” is a multiple choice concept question about the video. Videos and multiple choice problems are just two of the types of XBlocks allowed inside of the Verticals.

Finally, there are the children of Verticals. These XBlocks are actually displayed to learners and can be interacted with. There is a rather enormous branching factor in terms of the number of options and possibilities for the leaves of the course tree. At the time of writing, there are over sixty-five different XBlock options to go inside of units, each with a unique capability. To add to the complexity, some of these unit-level XBlocks, such as the Content Experiment XBlock, are able to nest more XBlocks inside of them rather than being a leaf on the course tree; however, the majority of the time, it is at this depth the course tree has leaves.

The most common types of XBlocks at the leaf level are videos, various types of problems, and HTML XBlocks. The different types of common problems include Checkboxes, Dropdown, Multiple Choice, Numerical Input, and Text Input. Examples of more complex problems that are seen in edX courses are Drag and Drop problems or in Computer Science courses, coding problems that will run the learner's code against a staff implementation to verify corrfectness.

3.2. Review XBlock

Now that the overall edX course structure has been explained, I will go into detail on the Review XBlock. It will be covered by going over the motivation behind creating the Review XBlock and then discussing the design of the XBlock and how it integrates into a course.

3.2.1. Motivation

The motivation behind the Review XBlock was to create a component that can integrate into edX courses to serve as a facilitator of spaced repetition. As stated in the Introduction and Related

Works sections, spaced repetition is a method of reviewing information after a period of time to reinforce the knowledge in the person's memory. My hope behind this XBlock is that it serves as a way to improve learning on edX by giving learners a chance to review the concepts from their courses in a simple manner.

To accomplish the goal of simulating spaced repetition inside edX courses, the Review XBlock was designed to display content from the course after different time periods. This content could be any sort of material from the course, including lecture videos and text, concept questions, and homework questions. In its ideal state, it would even be personalized on a per learner basis. This could lead to the Review XBlock only showing content to a learner if it has some belief that the learner has not fully mastered the material yet.

In implementation, the Review XBlock was simplified and would only focus on problems the learner had seen. The reason to limit to only problems was to not confuse learners by showing a lecture video from one section of the course with a problem from a completely different section. By only looking at problems, users could still review the concepts while staying in a single frame of mind of solving questions. In section 3.2.2., I will discuss some of the different design options considered along with their advantages and disadvantages before describing the final tool that was implemented and used inside of edX courses.

Another motivation behind the design of the Review XBlock was to ensure it does not require significant involvement on the course staff. As instructors are creating course content, managing hundreds to thousands of learners within their courses, and making sure everything runs as smoothly as possible, it was important to make the use of the tool only a small amount of additional work. While I succeeded in some aspects of this, such as enabling my tool to select the content to review without any instructor supervision, it is still an involved process to activate the XBlock within a course. Once activated however, it is simple to add in any number of Review XBlocks throughout the course.

3.2.2. Design

There were several possible designs that were considered for the XBlock. In the end, the Review XBlock became a mixture of two other edX XBlocks: the Randomized Content Block and the LTI Consumer. To give context before describing the implementation of the Review XBlock, the features of these two tools will first be explained.

The Randomized Content Block utilizes an edX data structure called a Content Library to randomly display a selected number of pieces of content. Content Libraries are data structures that store collections of XBlocks, such as videos, HTML blocks, or problems. In order to use a Content Library, the instructor for the class first creates one using the edX interface and associates it with an organization (i.e. MITx or HarvardX) and then gives it an identifier. Next, the instructor will upload however many components desired into the library so they are accessible inside of the courses.

To access the material inside of a Content Library, a member of the course staff would put a Randomized Content Block inside a Vertical in the course with a reference to the Content Library identifier. Then the course staff could specify the number of elements the Randomized Content Block should show and could even customize to only have a specific type of problem (Multiple Choice, True/False, Numerical Input, etc.) shown to the learners. When viewed by people in the course, the Randomized Content Block would go through its associated Content Library and display the desired number of problems to the learner from the Library randomly. Each learner who interacts with the block receives their own random content.

The Randomized Content Block offered many advantages. It had the ability to have a collection of content and display it to learners. It could also be placed multiple times in the course, which I believed would help in giving the spacing effect. However, there also seemed to be a couple of possible difficulties in using it. For one, I would have to decide what content should go inside of the Library. The simplest idea would be to just put all possible course content inside of it, but then I would have to switch up the selection algorithm so it would not potentially show content that has not been released yet. Additionally, if I was using it completely randomly and had lecture videos as well as problems are stored in a single Library, it could be an awkward user experience to receive a video from one lecture and a problem from a completely different part of the class so it loses cohesiveness. Alternatively, it could be possible to create multiple Libraries per topic or per week, but that would quickly snowball into an unwieldy size.

In the end, Randomized Content Blocks failed as a solution for a reason that was related to my perceived issues with it. Due to the way edX courses are constructed, all courses must have identical content for every learner who is in the course. Randomized Content Blocks seem to break that rule since each learner who sees it receives their own random content selected from the Content Library. The way edX handles this is it actually stores the entire Content Library in the location of the Randomized Content Block and then will display or hide problems based on which were randomly selected. The unfortunate part of this infrastructure is that if I were to create a Library that contained all of the content of the course (for example, 500 total XBlocks), every single Randomized Content Block put into the course would add an additional 500 XBlocks to the overall course structure. Since the idea is to have the reviews spaced out, there would need to be several (four or more for small courses and upwards of fifteen for large courses) of these blocks in the course. The end result would be making the course several times larger and since some of edX's internal operations run tree searches on the course tree, anything that would add hundreds or thousands of XBlocks to a course was infeasible. After discovering this fact, I dropped the idea of using Randomized Content Blocks and moved onto the next idea: hosting the review externally and connecting it to the actual course using edX's LTI (Learning Tools Interoperability) integration capabilities.

Learning Tools Interoperability, or LTI, is a way of linking education applications from a LTI provider to a LTI consumer. The standards that govern how LTI should operate is done by the IMS Global Learning Consortium ("Learning Tools Interoperability"). LTI providers are typically third-party websites that host educational content and have functionality to integrate into an LTI consumer. LTI consumers are able to utilize the providers by pulling in the provider's content into their interface and allowing users to interact with the material. An easy way to think about LTI is that the provider acts like a server and the consumer acts like a client. The client authenticates to the server and then can interact with problems that the server hosts. When the client submits a problem, it is sent to the server that will do any processing and grading necessary and then send back a response. Figure 5 shows an example of an edX LTI Consumer.

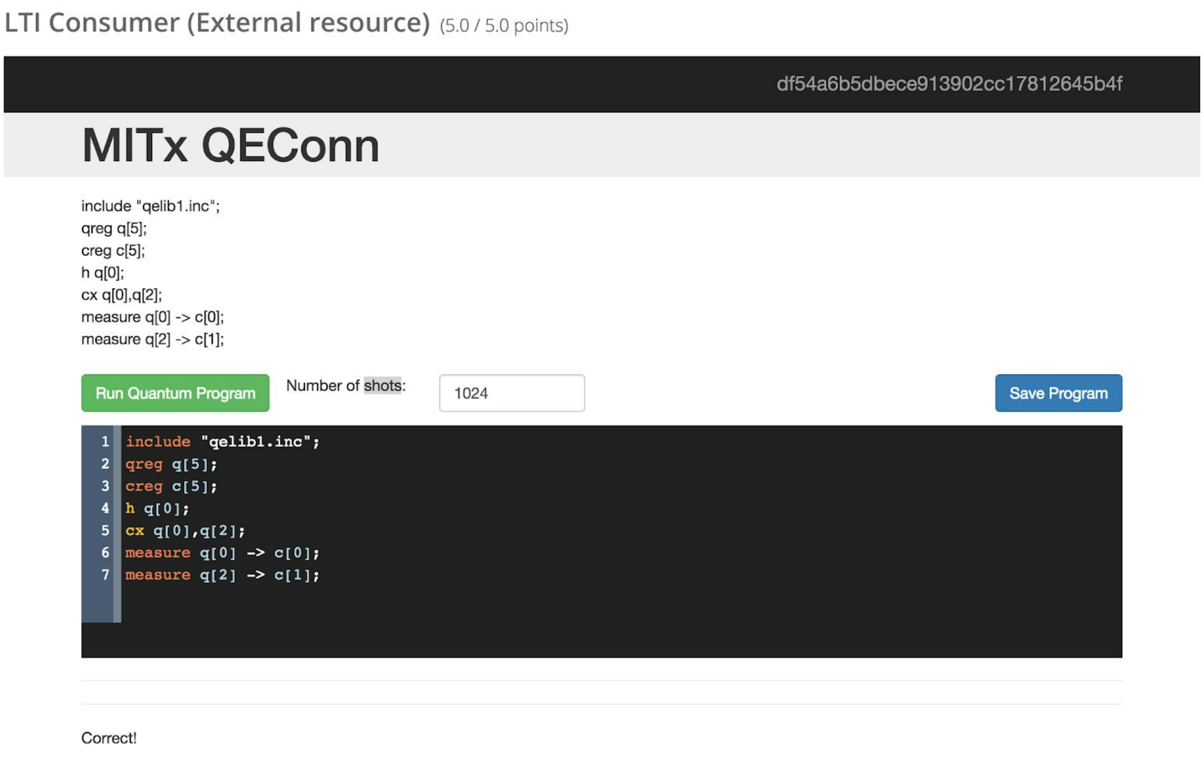

Figure 5. EdX LTI Consumer Example. Digital image. EdX Courseware. N.p., n.d. Web. 1 May 2018.
The image shows an example of an LTI Consumer in the edX platform. In this problem, the provider is showing a coding problem that the learner needs to solve. When the learner clicks "Run Quantum Program," the learner's code is sent to the provider which grades it and sends a response back.

Most Learning Management Systems, such as edX, are capable of acting as a LTI consumer. When acting as a consumer, edX inserts an iFrame into the LTI Consumer XBlock that displays the content from the LTI provider. EdX also has the ability to act as a LTI provider, which gave me the idea of using edX as both the provider and the consumer in order to create review content. This hit a roadblock because the production instance of edX does not fully support being a LTI provider and only a more experimental instance, edX Edge, has the functionality to be a provider. After speaking with several software engineers at edX, it was determined that using Edge as the provider and production edX as the consumer would be complicated and cause issues. The major complication would have been creating a communication interface between the two different instances of edX and then transmitting Personally Identifiable Information (such as learner usernames, emails, etc.) across this communication channel. Since Personally Identifiable Information always needs to be treated with care, the idea of using the LTI interface of edX was discarded and the Review XBlock was created.

The Review XBlock is a leaf XBlock in the course tree that is capable of simulating spaced repetition in edX courses. It does this by showing learners five problems they have previously interacted with in the course in a "sandbox" setting, meaning there are unlimited attempts, the review problems are ungraded, and they have the ability to see the answer after an attempt has been made on the problem. The sandbox setting for the problems was necessary so learners are able to feel as though they are actually reviewing content and not simply being tested again. Having the option to check for the correct answer is also helpful as it truly allows the learner to review regardless of whether they recall or not.

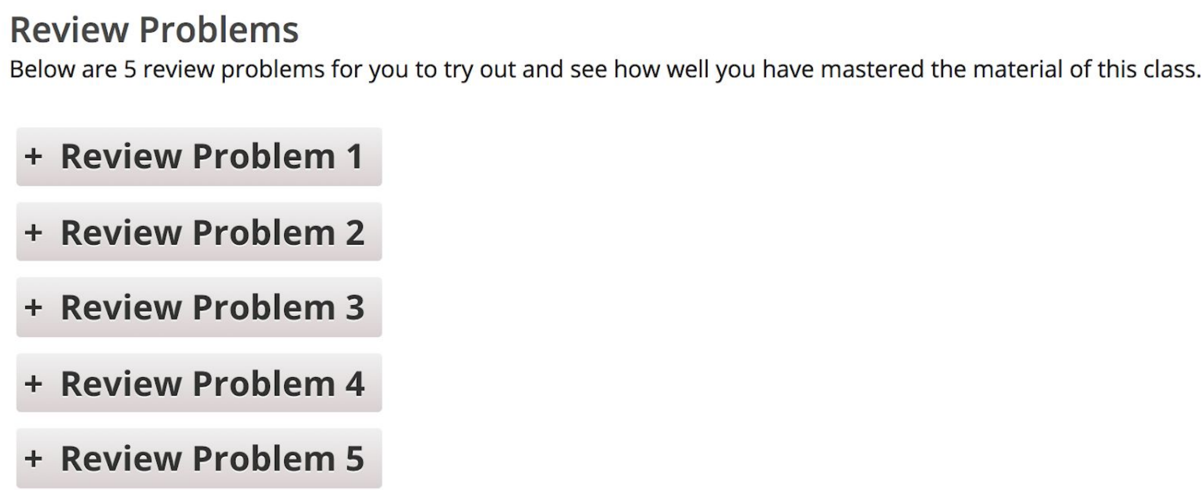

Figure 6. Screenshot from "MITx: 8.370.2x Quantum Information Science I, Part 2. Week 2 Review Unit." — Review XBlock.
This image shows an example of the Review XBlock being used in a course. The XBlock looks through all problems the learner has encountered in the course and then randomly selects five of them to display as review. The idea is that by reviewing problems from the course, learners are then reviewing the concepts associated with those problems.

The major feature of the Review XBlock is that it looks up problems for learners to review that they have encountered before. It does this through an edX datatable called the Courseware Student Module. The Courseware Student Module contains information such as what type of XBlock was encountered (Course, Chapter, Sequential, Vertical, problem, video, HTML, etc.), by whom (learner id), what course it was seen in, state, and more. It is important to note that information is only stored in the Courseware Student Module once a problem has been loaded by a user. Before a user can access a problem, the datatable will have no record of the XBlock in association to the user. When the Review XBlock is loaded by a user, it passes in a query to the Courseware Student Module asking for all problems the learner has seen in the current course. From there, it passes on the state field to the remainder of the backend logic.

The state of a problem is important for many reasons as it contains helpful information. Crucial fields of the state are the number of attempts a learner has used towards a problem and the score the learner has received on the problem as well as the total score the problem was out of. These are used in determining the eligibility of a problem to be displayed to a learner as well as correctness for the learner's performance on the problem. When displaying review problems to the learner, the Review XBlock uses these fields to let the learner know whether they correctly or incorrectly answered the problem originally and after how many attempts. An example of this can be seen in Figure 7.

As mentioned above there are certain criteria that problems need to meet in order to be eligible to be shown. The major reason for eligibility criteria is so that learners are not able to view graded problems (such as problem set questions or quiz material) after they are released, but before they are due. The largest deterrent to learners being able to see problems that they should not is the fact that the Courseware Student Module will not contain any record of XBlocks the learner has never seen before. This means that in courses that are set up with specific release dates for material (a common feature in instructor-led courses), a learner will not be able to see material from a future part of the course until the instructor releases it. Since the Review XBlock only looks at the Courseware Student Module for even possibly eligible problems, all problems in the future are automatically ruled out. Out of the remaining problems, there are four possible conditions for a problem to be eligible to be shown as review to a learner (at least one must be true):

1. The problem was originally ungraded. Since all review problems are also ungraded, it is safe to show the problem again as it will not affect the learner's grade.
2. All attempts on the problem have already been used. Although the problem is a graded problem, once a learner has used up all of their attempts, they can no longer add submissions to increase their grade. Since they can no longer take action to impact their grade, the problem becomes safe to show as review material.
3. The problem is past its due date. Similar to point two, after a problem's due date, a learner cannot take any action on that problem so it is safe to show it again as review.
4. The problem has already been correctly answered. If a learner has already answered a problem correctly, then there is no reason to prevent it from being use as review.

The Courseware Student Module gives us a way to find problems and then there is criterion to establish if a problem is eligible to be shown as review to a learner. Next comes displaying the problem to the learner. The Review XBlock features two ways to display problems to the learner. The first is the default and is focused on individual problems in a course. This method can only be used when problems are self-contained and are not reliant on other surrounding material. When this is the selected method of display, the Review XBlock will randomly select five of the eligible problems and will display those to the learner. I believe this is the more ideal form of display because it enables the learner to see five problems from anywhere they have encountered in the course, thus giving a diversity in their review. An example of what the Review XBlock looks like in the problem view is shown in Figures 6 and 7.

The second display the Review XBlock has is to show an entire Vertical to the learner. This is the desired behavior when problems rely on information surrounding them. An example of when this happens is when there is an HTML XBlock at the top of a Vertical that gives context to the problem, such as an image. The HTML block is followed up with several problem blocks that require knowledge of the image, but if the Review XBlock only displays the problems, they would be incomprehensible. In this case, it is necessary to show the complete unit to the learner so they are able to maintain context. When the Vertical display is required in a course, the Review XBlock randomly selects a single Vertical rather than randomly selecting five problems as in the default. Sadly, it is only desirable to show a single Vertical since it is unknown how many problems could be contained inside of the unit and it is essential the size of the review does not scare off any learners.

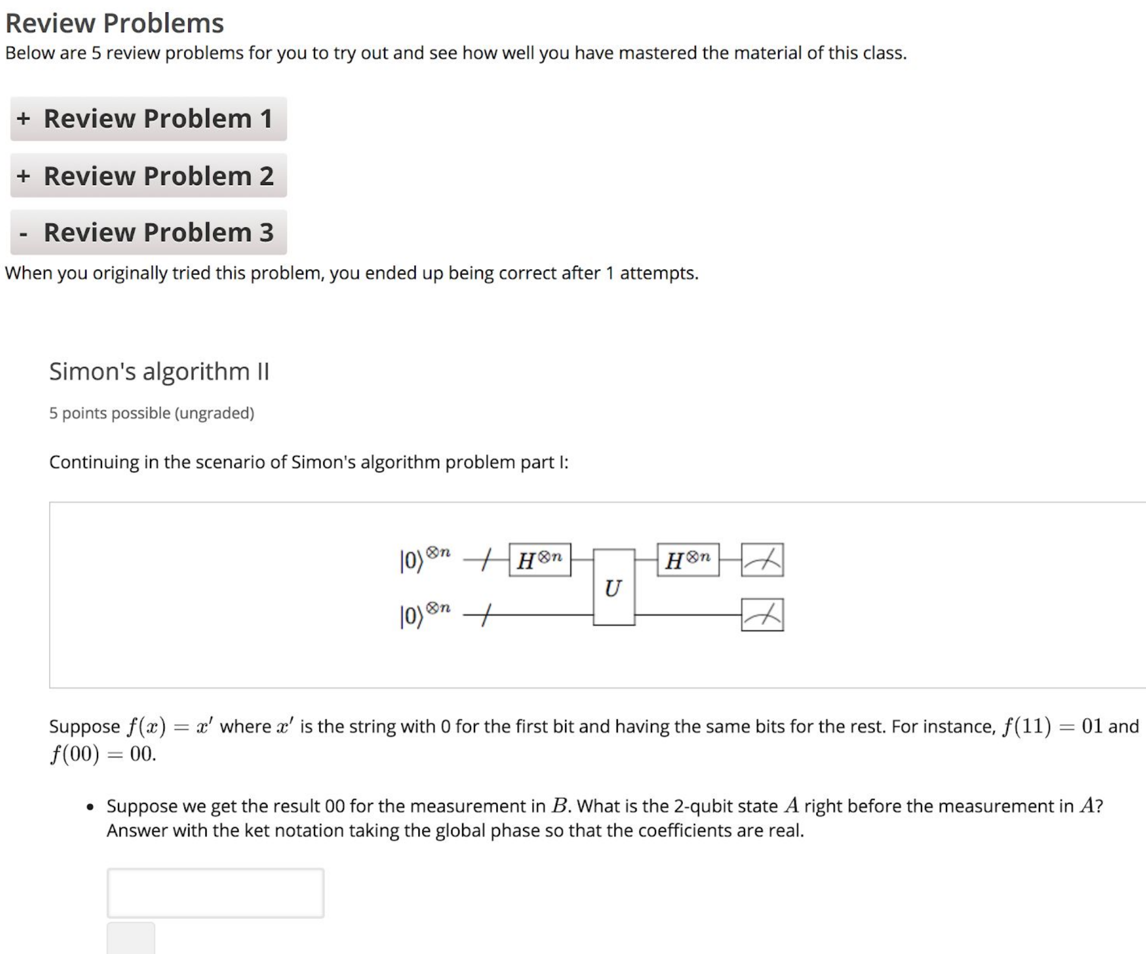

Figure 7. Screenshot from "MITx: 8.370.2x Quantum Information Science I, Part 2. Week 2 Review Unit." — Review XBlock Expanded.
This image depicts the Review XBlock in the problem view with one problem expanded. Two key insights are of the picture are that directly underneath the "Review Problem 3" button, there is text indicating how the learner originally performed on the problem ("correct" or "incorrect") and after how many attempts, and directly next to the number of points possible on the problem, it is specified that this problem is ungraded. Regardless if they problem was originally graded or not in the course, in the Review XBlock, all problems are ungraded.

Now that the implementation of the Review XBlock has been explained, I will quickly evaluate it as it relates to spaced repetition. Recall that in spaced learning, a person wants to at first review material after only a short timespan, and then after subsequent (successful) reviews, the frequency drops down until pure memorization is achieved and reviews stop. On an individual level, the Review XBlock fails at this task as it only shows a handful of problems to review and shows no preference to any problems.

However, when used throughout the course, the Review XBlock is able to offer a pseudo-spacing effect for the learners. Because the Review XBlock pulls from all problems a user has previously seen, in the beginning of the course, it has very few problems to choose from so the likelihood of reviewing a problem the learner recently encountered is high. As the course progresses and the learner views more problems, the probability of any given problem showing up also decreases. This means that for the early course material, the Review XBlock does a decent job of simulating the spacing effect (minus accounting for correctness) as the likelihood of seeing a problem falls as the course (and thus time) has progressed. Unfortunately, the same does not hold true across all problems in the course. Since every problem encountered has an equally probable chance of being chosen, at the end of the course, the problems seen are just as likely to come up in the Review XBlock as the ones from week one. This goes against the spacing effect as new information is not being prioritized for review. This limitation cannot be overcome in the current implementation of the Review XBlock, but since the XBlock allows infinite amount of review of all problems, a learner could continue to refresh an XBlock to load different problems to review. This at least allows for as much review as desired for a learner, even if the content of the reviews are not spaced out ideally.

3.3. Shadow Course

So far, I have only specified that the Review XBlock is able to display problems, but did not go into how it is able to actually show any problems. A reasonable assumption would be that since all review problems being displayed to learners are simply actual problems in the course, the Review XBlock would just pull in the already created instances of the problem and show them to the learner again. This approach leads to several issues. The first is that since it would be the same instance of the problem, all previous state of the problem is still captured in the XBlock. This means any answers previously submitted are still stored and shown to the learner; so, if the learner already correctly answered the problem, the correct answer would still be shown making it a rather useless review. The next logical thought I had to solve this problem is to hide or remove the state of the problem so the learner's previous submission is not shown. This led to an issue about grades because the XBlock's state stores the learner's grade for the problem so changing the state of the problem also changes the learner's grade. Clearly that is undesired behavior and so the solution of using the same instances again had to be discarded.

In section 3.2.2., I explained one of the issues with the Randomized Content Block was it would have to copy all contents of the Content Library into the location and this would lead to a course with additional hundreds or thousands of XBlocks. For the Review XBlock, this problem had to be circumnavigated so I came up with an idea: if the problem of the Randomized Content Block was creating too many blocks in the actual course, why not store them in a separate course? This led me to the idea of a *shadow course*.

The shadow course was created to function as a storage of review problems for the Review XBlock. This is accomplished by essentially creating a copy of the actual course with a couple of modifications to achieve the desired traits for review problems. By creating brand new instances of the problems in a new course, it is possible to change all problems to ungraded, remove any due dates, set the number of attempts to unlimited, and allow learners to show the answer after a single attempt. The shadow course thus offered many desirable traits. Some additional setup for the shadow course was also modifying the course identifier in a slight way from the original course so it is able to know what the associated shadow course is for an actual course.

Now there are two courses and an XBlock that is able to link them using course identifiers. The final component is how to actually show the problems from the shadow course to the learners within the Review XBlock in the actual course. This is accomplished in a way similar to the LTI Consumer XBlock where the shadow course acts as a pseudo-LTI provider and the Review XBlock acts as a pseudo-LTI consumer. The reason it is only a pseudo relationship rather than actual LTI is because the Review XBlock does not contain much of the actual protocols necessary for LTI, such as anonymizing users from the consumer to the provider. Inside of the Review XBlock, iFrames are used to display the problems from the shadow course. Every XBlock has a specific identifier and the Review XBlock stores these when doing its search through the Courseware Student Module. Using these identifiers, it is possible to display single problems (when in problem view mode) or entire units (when in vertical view mode) since both are simply XBlocks in the edX architecture. In Figure 7, it is possible to see a rendered iFrame containing a problem that is actually hosted in a separate course from the one it is being accessed by the learner. Behind each button in Figures 6 and 7 are individual iFrames, each containing a reference to a different problem. Although the Randomized Content Block and LTI Consumer XBlocks did not work for my research project, they both contributed to what would become the Review XBlock.

4. Experiment Setup

To test if the Review XBlock was helpful to learners, I set up an experiment in three edX courses: Quantum Information Science 1, Part 1 (8.370.1x), Quantum Information Science 1, Part 2 (8.370.2x), and Circuits and Electronics 3: Applications (6.002.3x). In order to divide the learners, I used an edX tool called a Content Experiment. A Content Experiment allows course staff to create different group IDs and will then randomly assign learners to one of the groups as soon as they come in contact with the Content Experiment XBlock. For my experiment, I created two groups: the *Control* group and the *Review* group. The Content Experiment would evenly divide learners who came in contact with it so it created a fifty-fifty split for the experiment. The display for the Review group is shown above in figures 6 and 7 and the display for the Control group is shown in figure 8.

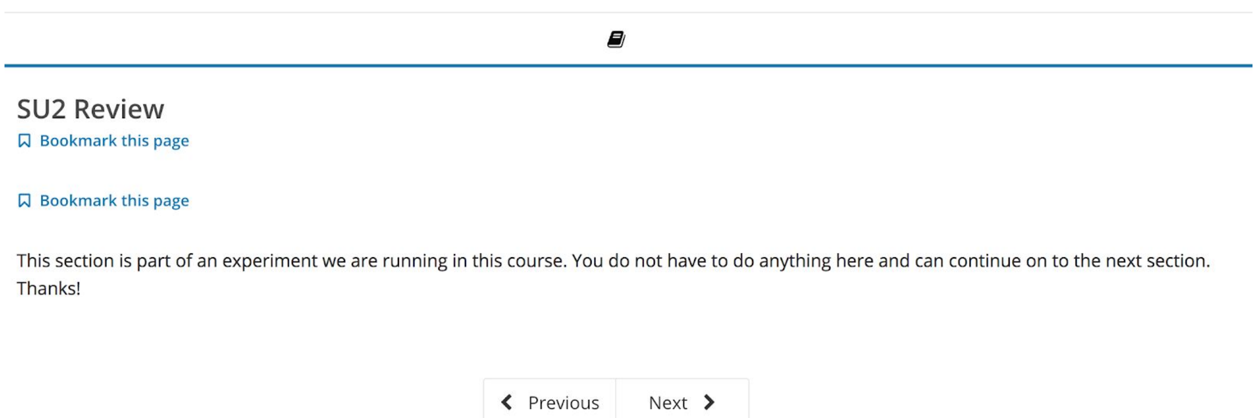

Figure 8. Screenshot from "MITx: 8.370.2x Quantum Information Science I, Part 2. Week 2 Review Unit." — Control Group View.
When interacting with the Content Experiment XBlock as part of my experiment, learners are divided into one of two groups: control or review. The Review group is shown the Review XBlock and a series of problems to use to review concepts from the course. The Control group, on the other hand, is shown a brief message (shown above).

The goal for the experiment was to allow learners to proceed through the course as usual with some of the learners having access to these Review XBlocks. Then, looking at different assessment scores for learners in the Control and Review groups, it will be possible to determine the quantitative effectiveness of the Review XBlock on learners. The different assessments in the Quantum Information courses are short concept questions after each video in a lecture and longer problem set questions at the end of each week. The Circuits course featured ungraded questions in each week's lecture content to test understanding from the lecture videos. For graded assignments, the course also had weekly homework assignments that would be included in a single vertical and feature a couple of problems and labs where users would typically use an edX circuit tool to build a desired circuit. By looking at a learner's grades on assessments and the number of events a learner triggered inside the course, I hoped to draw connections between the addition of the spaced review and scores on course material as well as engagement in the course. The methods used to analyze the data and results found will be discussed in section 5.

5. Experiment Analysis and Results

In looking at the collected data from the three courses I interacted with, there were some surprising results in terms of the number of users who interacted with the Review XBlock. The Quantum Information Science, Part 1 course had 29 users that made attempts on both review problems and actual problems out of 689 users assigned to the experimental group (4%). In the second part of the Quantum Information Science, the numbers dropped significantly. The number of learners that submitted a problem in the Review XBlock was down to 3 out of 200 possible users in the Review group (1.5%). The Circuits course had zero users submit any problems in the Review XBlock, which was possibly caused by its lower usage since it is a self-paced course that has been running for almost two years. In the content below, I am going to be discussing the different metrics I used to evaluate the review tool, but because of the low number of users in Quantum Information Science, Part 2 and Circuits and Electronics 3, my analysis will be restricted to only Quantum Information Science, Part 1. The tables used were constructed using Pandas in Python and the analysis was done in Stata.

As stated above, users were first put into either the control or experimental group. From there, I subdivided the groups into those who submitted problems to the Review XBlock and those who did not. The results of these four subpopulations can be found in Table 1. It is important to note before going into the bulk of the data analysis that since the relevant population (users assigned to the experiment and attempted review problems) only contains twenty-nine members, the statistics being generated have a slight bias due to the comparisons being between twenty-nine users and the remaining 1,345 users (users that never attempted any review, regardless of their experiment assignment). Whenever such a small sample size is being analyzed, it is possible that the results can end up misleading since there is not a large enough set to normalize out the results across all users. This can bias the final numbers being considered because the other groups being considered have a sufficient number of users and thus are more indicative of the standard user.

	Never submitted review problem	Submitted review problem	Total
Control group	685	0	685
Review group	660	29	689
Total	1345	29	1379

Table 1. Subpopulations from Quantum Information Science I, Part 1.
The table shows the number of users in Quantum Information Science I, Part 1 as broken down into their Content Experiment groups and if they submitted review problems. As expected, the Control group had zero users submit review problems and the Review group had twenty-nine out of the 689 possible.

5.1 Descriptive Statistics

The first types of analysis I looked into was simple descriptive statistics. This involved looking at different variables in the course and seeing if there were any surprising differences based on which of these subpopulations the users fall into. Naturally, the first one examined was the overall course grade. To start, I looked at the difference in course grades purely based on the experimental group the users were put in. The results can be seen in Table 2. When looking across the two groups, the average grade was comparable, showing no significant increase from having the review users in the review experimental group. I think this was to be expected since the review users only make up four percent of the total experimental group. As a result, even if the review users performed significantly above average or below average, when being included with the other 660 users in the Review group, their results are overwhelmed and their effect is not as easy to discern.

	Mean	Std. Err.	95% Conf. Interval	
(Control group)	.2641168	.0128808	.2388485	.2893851.
(Review group)	.2593614	.0127274	.2343942	.2843286

Table 2. Average Course Grades across Experiment Groups.
This table takes the average grade from the course as computed by edX and separates the result by the experiment assignments. The average grades tend to be low for courses because of the low completion rate in MOOCs leading to many users that will sample a course and then stop.

In order to isolate the review users, I again computed the average course grade, but broke it into the four subpopulations shown in Table 1. The results of this new analysis can be seen in Table 3. The average course grade for those users that submitted review problems was about twice as high as those who did not or could not. This is a positive result, but still must be conditioned on the fact that it is only looking at twenty-nine users as opposed to the 650+ in the other subpopulations. As a result, it is possible that these findings could just be noise in the course grades caused from only being able to look at a small sample set of review users.

	Mean	Std. Err.	95% Conf. Interval	
(Control group, Never submitted review problem)	.2641168	.0128808	.2388485	.2893851
(Review group, Never submitted review problem)	.2486212	.0128041	.2235035	.273739
(Review group, Submitted review problem)	.5037931	.0672423	.3718843	.6357019

Table 3. Average Course Grades across Experiment Groups and Submission of Review Problem.
Like Table 2, this table is looking at the average course grade with the addition of filtering the Review group by whether they actual used the review or not. The population that used the review tool seems to have a much higher average than the other groups, although this is due in part to not being weighed down as much by users who only try a few problems and then stop participating in the course.

In the same style of analysis, I also looked into overall engagement of users by the total number of events they triggered in the course. Examples of events that are counted are playing/pausing a video, checking if a problem is correct, saving a problem, showing the answer to a problem, and navigating through the course. I wanted to see if there could be a correlation between using the Review XBlock and overall engagement in the course. This could be caused by learners wanting to go and rewatch videos or review over lecture content after encountering a review problem on the same topic. See Table 4 for the number of events analysis by experiment groups and Table 5 by experiment group and submitting a review problem subpopulations. In a similar fashion to the course grade, when comparing over the evenly split experiment assignments, there is not a large

	Mean	Std. Err.	95% Conf. Interval	
(Control group)	1435.785	76.4405	1285.833	1585.738
(Review group)	1713.598	191.7015	1337.538	2089.685

Table 4. Average Number of Events across Experiment Groups.
This table computes the average number of events per user based on their randomly assigned group in the experiment. The Review group has the higher average, but with a higher standard error leading to the groups being roughly similar with only a slight bias to the Review group.

	Mean	Std. Err.	95% Conf. Interval	
(Control group, Never submitted review problem)	1435.785	76.4405	1285.833	1585.738
(Review group, Never submitted review problem)	1650.777	199.0097	1260.381	2041.173
(Review group, Submitted review problem)	3143.31	404.4146	2349.973	3936.648

Table 5. Average Number of Events across Experiment Groups and Submission of Review Problem.
When filtering out the users that submitted the review problems, it is clear that they hold a significant increase over the remaining users. Focusing on the confidence intervals, it is easy to see how even the lower end of those that used the review problems had considerably more events logged.

difference between the two groups. The Review group has a higher mean by almost 300 events, but the standard error is over twice as large as the Control group. Table 5 however sees the two groups that did not use any review problems grow closer (only a 200 event difference now), with the users that used the review take the definitive lead with nearly double the number of events. Although the standard error is large for the users that submitted review, the lower bound of their 95% confidence interval is still greater than the upper bound of the other groups interval by 200 events in the closer of the two subpopulations.

Next, I wanted to look into the characteristics of users who actually used the review tool compared to the rest of the course. To do this, I used the three subpopulations of the control group and the experimental group split on if they had submitted a review problem and then looked at the averages of other columns in the dataset. The other variables I looked into are whether the learner earned a certificate in the course (requires completing the course with a minimum percentage), if they indicated they held a bachelor's degree or higher, if their profile said they are located in the United States, their gender (male or female only), and their calculated age in 2018 if their year of birth was given. The tables with the results are in Appendix A, and I summarize the results below.

For simplicity, for the remainder of this section, I will call the Review group that did not submit any problems Review 0 and the Review group that did submit problems Review 1. The overall certified rate in the course was 8%. When broken down into the subpopulations, the Control group had a 7.89% and the Review 0 group was at a 7.88% certified rate. These are both to be expected as they are nearly the overall rate and include 96% of all users. The Review 1 group though had a 13.8% certified rate.

The remainder of the considered variables do not include all possible users in the course since they rely on information the user provides in their edX profile. Rather than comparing the 1,379 users that interacted with the course, the bachelor’s degree variable only looks at 924 user, the USA variable considers 1,093 people, the gender variable looks at 964 learners, and the age variable uses 932 observations. The bachelor’s degree or higher variable resulted with 78.5% of users in the course having a bachelor’s or higher. The Control group had 78.9%, the Review 0 group had 77.4%, and the Review 1 group had 90.5%. The percentage of users located in the USA was at 30.6% through the course. Review 1 group in this case was the lowest at 26% indicating that even more than the average in the course, the review users were international. The gender and age variables both came out similar across all subpopulations. The course had 5.6% females and each group was within 1.3% with the Review 1 group at 4.3%. The average age in 2018 for users that provided it was 36.66 and the Review 1 group was again the furthest away at 38.3 years old.

Looking through all of the descriptive statistics, it is possible to get an idea of the typical user that used the Review XBlock. They are users that interact with the course more, are more likely to earn a certificate, are more formally educated, international, and are slightly older. Although these are ideas only coming from a single course, I believe some of these would likely hold true across all courses. More motivated users are more likely to utilize the extra review when offered and that can be easily seen by looking at the users with a high number of events and earning certificates. As such, it is possible that it is already a self-selecting group of individuals that have the desire to use the Review XBlock and thus the results reflect these types of users as opposed to the standard edX learner.

5.2 Multiple Regression Analysis

The last type of analysis I did on the course data from Quantum Information Science, Part 1 was a multiple regression analysis. Multiple regression analysis is used to learn more about the relationship between several independent variables and a dependent variable. The major output from the regression that I was looking at was the p-value for each independent variable. The p-value is used to establish statistical significance of a variable. To do this, a p-value must be below a designated threshold, and if it is, the associated variable is considered significant enough to reject the null hypothesis. The null hypothesis represents there being no relationship between two or more measured variables. In the scenarios considered below, this would mean the independent variables I am considering have no relationship on the dependent variable, the average course grade. I used the traditional threshold of 0.05 for the p-value and determining statistical significance.

For the purposes of my experiment, I performed three sets of regression, using the course grade as the dependent variable. In the first regression, I used age in 2018, education level, and being located in the US or not as the independent variables. The second regression added in if a user used the Review XBlock as an independent variable and the third regression built from the second by also adding in the number of problem checks to the analysis. These can be seen in Tables 4, 5, and 6, respectively.

Course Grade	Coefficient	Std. Error	t	p-value	95% Confidence Interval	
Age in 2018	.0001146	.0011873	0.10	0.923	-.0022167	.0024458
Education	.0323181	.0116371	2.78	0.006	.0094687	.0551674
USA	.0182107	.0301095	0.60	0.546	-.0409092	.0773307
_cons	.1152802	.052553	2.19	0.029	.0120926	.2184679

Table 6. Multiple Regression 1 evaluating Course Grade.
The first regression performed shows that the best indicator for the overall course grade is the learner's education level.

Course Grade	Coefficient	Std. Error	t	p-value	95% Confidence Interval	
Age in 2018	.0003268	.0017062	0.19	0.848	-.0030296	.0036831
Education	.0254944	.0162675	1.57	0.118	-.0065066	.0574954
USA	.0272331	.0425567	0.64	0.523	-.0564836	.1109497
Review	.2378478	.0821286	2.90	0.004	.0762862	.3994095
_cons	.124877	.0712742	1.75	0.081	-.015332	.2650861

Table 7. Multiple Regression 2 evaluating Course Grade.
The second regression performed shows that the best indicator for the overall course grade is actually if they used the review tool. With a p-value much less than 0.05, the review tool gives strong evidence against the null hypothesis.

Course Grade	Coefficient	Std. Error	t	p-value	95% Confidence Interval	
Age in 2018	-.0013716	.0011403	-1.20	0.230	-.0036148	.0008715
Education	.0179343	.0108489	1.65	0.099	-.0034077	.0392763
USA	.0152425	.028371	0.54	0.591	-.0405689	.0710539
Review	.0431645	.0555705	0.78	0.438	-.0661539	.1524829
Number of Problem Checks	.0020889	.0001027	20.34	0.000	.0018869	.0022909
_cons	.0563036	.047625	1.18	0.238	-.0373843	.1499916

Table 8. Multiple Regression 3 evaluating Course Grade.
The third regression performed shows that the best indicator for the overall course grade is the number of problem checks a user has done in the course. This result makes sense as users that are checking more problems are obviously completing more of the course. This will almost always end in a higher grade.

The regressions are valuable at being able to show which variables are able to closely predict the dependent variable. Without more telling information, the education level becomes the best indicator of performance in the course. Then, once the review variable is introduced, it is able to most closely foresee the course grade. Although the number of problem checks dominates the prediction for the course grade in the final regression; in its absence, the review tool is the best predictor out of the tested variables.

After going through all of the analysis, I believe it is safe to say that while use of the Review XBlock is suggestive of better performance and engagement, it is still too early to say if it would end up holding statistical significance in the grand scheme of the course. I believe the few number of users who ended up interacting with it severely limits the analysis able to be performed. It makes it easy to bias any of the variables being examined since it is comparing a handful of users against the entirety of the rest of the course. In section 6, I will discuss improvements I believe could be made to the Review XBlock and the experiment in general to help achieve better statistical significance in the future.

6. Improvements

The most obvious improvement for the experiment is to increase its scope in terms of the number of courses used. I was limited to using three courses, all of a rather short length. Some of the courses on edX run for twelve or fifteen weeks, while the courses I interacted with were all designed to take about four weeks. I believe a longer course would help increase the draw of having the option to review problems and would also give learners more opportunities to come across the review unit. In a similar vein, I believe there is some more experimentation that can be performed on the location of the Review XBlock in the course structure. For the Quantum Information courses, Review XBlocks were located at the end of a week after the problem set, which may have reduced their use because learners would only naturally come across them after finishing the mandatory assignments for the week. If there are past runs of the courses that are able to be compared against, I believe it would also help the experiment to open it up to the entire course rather than limiting the experiment population size by half from the very start.

There are several ideas to improve the Review XBlock. The first in my mind is to improve on the actual spaced repetition design and being able to control it within the XBlock rather than relying on course position. If the Review XBlock was able to have a reference to the number of times the user correctly answered a problem and when the user last submitted the problem in the actual course or as review, it could create a bank of review problems based on the last submission time and correctness. Then it could display problems that follow the Leitner System more closely by delaying problems recently answered correctly as opposed to the completely random approach taken now.

A second improvement is to better the personalization of the Review XBlock. This could be done in many ways, such as the one listed above. Other personalization modifications that could happen are letting the user specify the number of problems they would like to see or even narrowing it by a specific section they would like to focus on. Additionally, the Review XBlock could be configured to focus on problems originally answered incorrectly if the user or instructor desired. A potential option to try to improve the Review XBlocks usage is to advertise it to the course so learners are more encouraged to use it.

7. Conclusions

Over the course of the past year, I built a tool that is able to help simulate spaced repetition in edX courses, the Review XBlock. The Review XBlock is able to look at all problems a user has seen in a course and randomly select a few of them to show to learners. When displayed to a learner, it looks identical to when it was originally seen, but is now in a sandboxed setting. This means the user has unlimited attempts, it is ungraded, and the user can see the answer if they choose. In order to do this, the Review XBlock utilizes components of the Randomized Content Block and LTI Consumer by using a shadow course to host problems and iFrames to display them.

After implementing the Review XBlock, I ran an A/B experiment in three edX courses to evaluate its effectiveness. Although not widely used, the data suggests that the learners who do use it tend to perform well in the course and trigger more events, an indication of engagement in the course. As a result, I believe the experiment I ran was moderately successful and could be confirmed by rerunning the experiment in more courses. In doing so, it would be possible to gather more users across different courses and subjects and determine if the same findings presented in this paper hold in other courses.

Improvements could be made to the Review XBlock to increase its effectiveness for executing spaced learning as well as the personalization capabilities of the tool. Although not a perfect solution, I think the Review XBlock does an adequate job of simulating spaced repetition in an environment that allows learners to truly review course material without consequences. I believe it can only improve a learner's experience by including it in more courses on edX.

Appendix A

Tables

Certified	Mean	Std. Error	95% Confidence Interval	
(Control group, Never submitted review problem)	.0788321	.0103037	.0586194	.0990448
(Review group, Never submitted review problem)	.0787879	.0104946	.0582006	.0993751
(Review group, Submitted review problem)	.137931	.0651663	.0100948	.2657673

Table A-1. Average Percentage of Users who earned a Certificate across Experiment Groups and Submission of Review Problem.

Bachelors Plus	Mean	Std. Error	95% Confidence Interval	
(Control group, Never submitted review problem)	.7885835	.0187941	.7516993	.8254677
(Review group, Never submitted review problem)	.7744186	.0201795	.7348155	.8140217
(Review group, Submitted review problem)	.9047619	.0656383	.7759442	1.03358

Table A-2. Average Percentage of Users who hold a Bachelors or higher across Experiment Groups and Submission of Review Problem.

USA	Mean	Std. Error	95% Confidence Interval	
(Control group, Never submitted review problem)	.3154982	.0199796	.2762954	.3547009
(Review group, Never submitted review problem)	.2996183	.0200309	.2603149	.3389217
(Review group, Submitted review problem)	.2592593	.0859436	.090626	.4278925

Table A-3. Average Percentage of Users that are from the USA across Experiment Groups and Submission of Review Problem.

Gender	Mean	Std. Error	95% Confidence Interval	
(Control group, Never submitted review problem)	.0576132	.0105805	.0368497	.0783766
(Review group, Never submitted review problem)	.0549451	.0106946	.0339576	.0759325
(Review group, Submitted review problem)	.0434783	.0434783	-.0418448	.1288013

Table A-4. Average Percentage of Users that are Female (used as a binary indicator for gender) across Experiment Groups and Submission of Review Problem.

Age	Mean	Std. Error	95% Confidence Interval	
(Control group, Never submitted review problem)	36.53092	.6327724	35.28909	37.77274
(Review group, Never submitted review problem)	36.71818	.66498	35.41315	38.02322
(Review group, Submitted review problem)	38.30435	2.971827	32.47209	44.1366

Table A-5. Average Age across Experiment Groups and Submission of Review Problem.

REFERENCES

Abelson, Hal. "The creation of OpenCourseWare at MIT." Journal of Science Education and Technology 17.2 (2008): 164-174.

Chafkin, Max. "Udacity's Sebastian Thrun, godfather of free online education, changes course." Fast Company 14 (2013).

Ebbinghaus, Hermann. "Memory: A contribution to experimental psychology." Annals of neurosciences 20.4 (2013): 155.

EdX LTI Consumer Example. Digital image. EdX Courseware. N.p., n.d. Web. 1 May 2018.

Jordan, Katy. "Initial trends in enrolment and completion of massive open online courses." The International Review of Research in Open and Distributed Learning 15.1 (2014).

Kerfoot, B. Price, et al. "Online spaced education generates transfer and improves long-term retention of diagnostic skills: a randomized controlled trial." Journal of the American College of Surgeons 211.3 (2010): 331-337.

Khalil, Hanan, and Martin Ebner. "MOOCs completion rates and possible methods to improve retention-A literature review." World Conference on Educational Multimedia, Hypermedia and Telecommunications. Vol. 1. 2014.

"Learning Tools Interoperability" Learning Tools Interoperability | IMS Global Learning Consortium. N.p., n.d. Web. 01 May 2018. <https://www.imsglobal.org/activity/learning-tools-interoperability>.

"MITx: 8.370.2x Quantum Information Science I, Part 2." Course | 8.370.2x | EdX. N.p., n.d. Web. 30 Apr. 2018. <https://courses.edx.org/courses/course-v1:MITx+8.370.2x+1T2018/course/>.

"MITx: 8.370.2x Quantum Information Science I, Part 2. Week 2 Review Unit." SU2 Review | 8.370.2x | EdX. N.p., n.d. Web. 2 May 2018. <https://courses.edx.org/courses/course-v1:MITx+8.370.2x+1T2018/courseware/Week2/w2_review_sequential/1?activate_block_id=block-v1%3AMITx%2B8.370.2x%2B1T2018%2Btype%40vertical%2Bblock%40w2_review_unit>.

Murre, Jaap MJ, and Joeri Dros. "Replication and analysis of Ebbinghaus' forgetting curve." PloS one 10.7 (2015): e0120644.

Pappano, Laura. "The Year of the MOOC." *The New York Times* 2.12 (2012): 2012.

Taneja, Shilpi, and Anita Goel. "MOOC Providers and their Strategies." International Journal of Computer Science and Mobile Computing 3.5 (2014): 222-228.il

Zirguezi. Leitner System. Digital image. Leitner System. Wikipedia, 19 July 2012. Web. 22 June 2017.