Article

Darryl Chamberlain Jr.
University of Florida

Russell Jeter
Emory University

Creating diagnostic assessments: Automated distractor generation with integrity

The goal of this paper is to propose a new method to generate multiple-choice items that can make creating quality assessments faster and more efficient, solving a practical issue that many instructors face. There are currently no systematic, efficient methods available to generate *quality distractors* (plausible but incorrect options that students choose), which are necessary for multiple-choice assessments that accurately assess students' knowledge. We propose two methods to use technology to generate quality multiple-choice assessments: (1) manipulating the mathematical problem to emulate common student misconceptions or errors and (2) disguising options to protect the integrity of multiple-choice tests. By linking options to common student misconceptions and errors, instructors can potentially use multiple-choice assessments as personalized diagnostic tools that can target and modify underlying misconceptions. Moreover, using technology to generate these quality distractors would allow for assessments to be developed efficiently, in terms of both time and resources. The method to disguise the options generated would have the added benefit of preventing students from working backwards from options to solution and thus would protect the integrity of the assessment. Preliminary results are included to exhibit the effectiveness of the proposed methods.

Keywords: Assessments, Assessment Generation, Automated Item Generation, Distractors, Diagnostic Tools, Multiple-Choice.

1. Introduction

Assessment is a critical component of every course. There are two common types of assessment: summative and formative. S*ummative* assessments strive to record student achievement while f*ormative* assessments strive to gather evidence of student learning in order to modify instruction (Cauley & McMillan, 2010). In other words, the primary role of formative assessment is diagnostic – to inform the instructor what each student knows or does not know over some area of content. While there are numerous ways to assess students' knowledge, multiple-choice tests are the most widely used assessments in K-16 as they can be the most efficient to administer while simultaneously being quick and objective to grade (Rodriguez, 2011; Haladyna & Rodriguez, 2013). We use a typical College Algebra item to contextualize multiple-choice item terminology in mathematics.

[Stem]

Solve the linear equation below.

[Problem]

$$\frac{-3x-6}{3}-\frac{-8x-8}{5}=\frac{7x+6}{2}$$

[Options]

A. $x=\frac{-40}{29}$ **[Distractor]**

B. $x=\frac{-66}{29}$ **[Distractor]**

C. $x=\frac{-34}{29}$ **[Solution]**

D. $x=\frac{-17}{10}$ **[Distractor]**

Figure 1: Example of a typical multiple-choice item.

A *multiple-choice item* consists of a stem and options. The *stem* includes the context, content, and problem for the student to answer. In Figure 1's example, this includes the instructions (context) and the problem. By *problem*, we refer to the content issue that must be solved. In the example in Figure 1, this would be solving the linear equation. Solving this problem leads to the *solution*. Plausible, but incorrect, answers to the problem are referred to as *distractors*. The solution and distractors are used to create the *options*, or choices presented that the student must choose from.

Numerous guides for constructing quality multiple-choice questions exist and they largely agree on the best practices for developing assessments (Moreno, Martinez, & Muniz, 2015; Frey, Petersen, Edwards, Pedrotti, & Peyton, 2005). These guides are routinely used by content specialists to create multiple-choice items, which are then disseminated for general use. Guidelines commonly focus on writing the content and choices of an item. For example, Haladyna et al. (2002) proposed 31 suggestions when writing multiple-choice items: 8 related to content and 14 related to choices. These suggestions can be vague (e.g., "avoid trick or ambiguous items") and do not provide a way to systematically develop multiple-choice items. In fact, the authors state ``The science of MC item writing is advancing, but item writing is still largely a creative act" (p. 329). The development of a systematic guide to create distractors based on common errors and misconceptions would provide an avenue to advance multiple-choice item writing in a "non-creative" way.

2. Literature on Distractor Generation

Creating the stem, problem, and solution for a multiple-choice item in K-14 mathematics is a relatively straightforward task. Item content development follows the objectives laid out in the associated textbook, developed by the textbook author(s) to focus on specific content. No

such blueprint exists for developing the distractors though. For example, consider a question that asks students to expand the expression $(x-y)^2 = x^2 - 2xy + y^2$. A student with incomplete knowledge of polynomial expansion may choose $x^2 + y^2$ as the expansion and ignore the second term. Another student with partial knowledge of polynomial expansion may choose $x^2 - y^2$ and consider 'distributing the exponent' as a valid mathematical operation (Filloy & Rojano, 1989). These two examples illustrate common student misconceptions with polynomial expansions – misconceptions instructors want to capture during formative assessment so that these conceptions can be challenged and subsequently modified. This illustrates one of the biggest hurdles for creating quality distractors: the misconceptions a student may hold can be item-specific, requiring an item-by-item analysis. Without a systematic method to develop these distractors efficiently, creating a single assessment can be a timely endeavor.

It is well-known that distractors play a fundamental role in multiple-choice tests for any topic (Haladyna & Rodriguez, 2013). Gierl et al. (2017) consider distractors to (i) require a significant amount of time and resources to create, (ii) affect item quality and learning outcomes, and (iii) provide diagnostic inferences about students' knowledge (e.g., inferences about what students know or do not know). The authors go on to say that "Distractor development, in fact, is often considered by content specialists to be the most daunting and challenging component of writing a multiple-choice item" (p.1086). Yet, research on empirically-supported development of quality distractors for multiple-choice items is relatively sparse, even in the context of mathematics specifically (Gierl, Lai, Hogan, & Matovinovic, 2015). The following paragraphs will review the recent advances in generating quality distractors and how this paper will expand on these advances.

There are currently three general strategies to generate distractors (Author & Author, 2019). The first focuses on common misconceptions in student thinking while they reason about the problem. We illustrated this with polynomial expansion as students hold two pronounced misconceptions about polynomial expansion. These misconceptions can be recalled and utilized by experienced content specialists reflecting on the common errors they have seen in the past (Collins, 2006) or identified through evidence-based research on students' work during open-ended items (Briggs, Alonzo, Schwab, & Wilson, 2006). As such, this approach creates high-quality distractors that mirror mistakes (based on misconceptions) students may make during an assessment. This quality comes at a steep price – a great deal of time and resources must be used to develop these distractors, especially for items developed through evidence-based research (Gierl, Bulut, Guo, & Zhang, 2017).

The second strategy focuses on similarities between the solution and distractors. For example, a numeric solution such as $\frac{3}{4}$ could be manipulated in some form (e.g., being negated, divided by a factor, or divided by 1) to provide a host of distractors like $\frac{-3}{4}, \frac{4}{3}, -\frac{4}{3}$. In contrast to the first strategy, manipulating the solution in some way to make similar responses does not require a great deal of time and resources, and thus is commonly utilized (Gierl, Bulut, Guo, & Zhang, 2017). The disadvantage to this method is that distractors may not reflect actual mistakes a student would make on the assessment. Students with incomplete knowledge may be able to eliminate these types of distractors and thus arrive at the solution (or, at least, more easily guess at the solution). Alternatively, students who completed the problem correctly may accidentally choose visually similar distractors and thus their multiple-choice answer would not accurately reflect their knowledge. Due to these limitations, some authors have suggested multiple-choice assessments cannot provide diagnostic information (Lissitz, Hou, & Slater, 2012) rather than the

more nuanced position that multiple-choice assessments are not commonly written to provide diagnostic information.

The third strategy relies on utilizing research on how students develop an understanding of concepts to model student responses at different levels of conception. For example, the Precalculus Assessment by Carlson, Oehrtman, and Engelke (2010) utilized a theoretical model for how students develop an understanding of covariational reasoning, the Covariation Framework (Carlson, Jacobs, Coe, Larsen, & Hsu, 2002), along with interview-based research to create common student reasoning based on each level of understanding. These responses were used as the option choices for multiple-choice assessments that targeted students' level of understanding. Similar to the first strategy, developing interview-based items was a resource-heavy endeavor (Carlson, Oehrtman, & Engelke, 2010).

In short, creating distractors based on conceptions and/or misconceptions is preferred but not always feasible, and thus distractors are commonly developed based on small variations of the solution. One avenue for creating quality distractors based on conceptions and common misconceptions is *Automatic Item Generation* (AIG). AIG utilizes computer technologies and content specialists to automatically generate problems, solutions, and quality distractors. By *automatically*, we mean that an item structure can be developed ahead of time that some technology would use to create many items without the need for future human intervention. Few examples of AIG currently exist, even in the context of mathematics (Gierl, Lai, Hogan, & Matovinovic, 2015; Gierl, Bulut, Guo, & Zhang, 2017). We now review one of the most recent, relevant works in AIG to set the stage for our method. This method was written to be general and used medical science as the context for their examples.

Gierl and Lai (2013) described a three-step process for generating multiple-choice items. First, an *item model*, or the general scaffolding of the stem and problem, is developed. Then, the content knowledge required to solve the problem to be used in the item is determined. Finally, computer-based algorithms are used to place content from step 2 into the item model from step 1. The authors suggest that "Using this three-step process, hundreds or even thousands of items can be generated using a single item model" (p. 37). We provide a short overview of each of these three steps as described by Gierl and Lai (2013), along with our own examples in a mathematical context.

Step 1: Item Model

There are currently two types of item models: 1-layer and n-layer item models. A 1-layer item model manipulates some small number of elements in the model, all at the same level. We can think of this as choosing 1 element from some set. For example, to generate a linear equation of the form $y = mx + b$, we could choose m and b to be rational numbers. Choosing a single rational pair (m, b) provides the single set needed to change the mathematical problem at hand. This would generate a 1-layer item asking students to solve the equation $y = mx + b$ 1-layer item models are ubiquitous in current multiple-choice tests (Gierl & Lai, 2013).

An n-layer item model manipulates many elements at multiple levels in a model. We can think of this as choosing 1 element from numerous sets. For example, an item model may ask students to solve a linear equation of *any* form. The item model could first choose the form the linear equation would be displayed in (e.g., standard, point-slope, slope-intercept). The item model could also choose the types of numbers that would be used in the linear equation (e.g., Naturals, Integers, Rationals). This would create a 2-layer item that chooses one element from the sets of equation type and number type. After making these two choices, the problem

equation can be generated. In summary, the *n*-layer structure has multiple layers of elements, where each element can be varied *simultaneously* to produce varying items. The n-layer model can thus quickly develop test items that address many content objectives based on how the elements in the structure are chosen, though care needs to be taken to ensure consistent item difficulty. This will be addressed in step 2.

Step 2: Content

After determining the item model, content specialists are used to identify the content. Two general approaches to identifying content exists: weak and strong theory (Gierl & Lai, 2013). *Weak theory* uses design guidelines to create new item models that remain similar (in terms of difficulty and structure) to the original item model and is commonly employed in 1-layer item models. For example, to create similar linear equations to solve, the content specialist would choose a single type of linear structure and a single type of elements for this structure, as introducing additional changes may fluctuate the difficulty of the item. That is, the linear equation $4 = 2x + 5$ may be easier for a student to solve than the equation $\frac{-3x-6}{3} - \frac{-8x-8}{5} = \frac{7x+6}{2}$ as they utilize different structures of a linear equation. Similarly, the linear equation $\frac{4}{7} = \frac{2}{3}x + 5$ would likely be more difficult for a student than $4 = 2x + 5$ as it introduces rational numbers to the same structure. This illustrates why 1-layer items are ubiquitous in assessment generation.

Strong theory utilizes a cognitive model to identify and manipulate items that may change the difficulty level of the item. While relatively few cognitive theories exist to guide general item development practices (Gierl & Lai, 2013), many have been proposed in the last 30 years in undergraduate mathematics education (Leatham, 2014). These can be utilized to model the knowledge and skills a theoretical student may need to solve the mathematical problem, which in

turn can provide guidance to develop item models and manipulate the elements of the item model. This potential was illustrated in the Precalculus assessment by Carlson, Oehrtman, and Engelke (2012).

Step 3: Computer-Based Algorithms

Once the item model is created and the content for the model determined, a computer program is needed to assemble the two to create specific items. While software has been developed specifically for generating test items, Gierl and Lai (2013) state "… it is also important to note that any linear programming method can be used to solve the type of combinatorial problem found within AIG" (p. 43-44).

The three-step method above focuses on item generation holistically. Gierl and Lai (2013) showcased an n-layer structure with a possible solution list that remained static while the stem was changed, resulting in different solutions from the static solution list. The resulting distractors were the rest of the possible solutions, which may or may not have mirrored student misconceptions based on the randomly generated problem. This illustrates how the n-layer structure does not inherently describe how distractors could be automatically generated based on student misconceptions *for the particular problem generated*. As generating distractors is the most difficult aspect of multiple-choice item generation (Gierl, Bulut, Guo, & Zhang, 2017), we will introduce a novel method to automatically generate distractors by manipulating the *problem* within the stem in a way that reflects students' misconceptions and mistakes. The following section details this distractor-generation process.

3. Automated Assessment Generation Method

We present a method for generating assessments that is grounded in the idea of creating *nearby problems* based on common errors made while solving the original problem as well as on common misconceptions students have with the content being evaluated. From these nearby

problems, one can create a set of *distractor solutions* that can be used as answer choices in a multiple-choice item.

3.1 Question and Solution Generation

Before we can discuss the process by which we create *plausible* distractors, the reader must have a clear understanding of how questions can be randomly generated, and by extension, the solutions (correct answers) to those questions. Figure 2 introduces the sample question that we use to walk the reader through the methodology for the automated assessment algorithm conceptually, before presenting the algorithm more generally.

Question 1. Solve the linear equation below.

$$\frac{-3x-6}{3}-\frac{-8x-8}{5}=\frac{7x+6}{2}$$

A. $x=\frac{-40}{29}$

B. $x=\frac{-34}{29}$

C. $x=\frac{-66}{29}$

D. $x=\frac{-17}{10}$

Figure 2: College Algebra example item.

A question of this type can be randomly generated from a template for questions that involve solving rational equations. To create this template, all coefficients in the numerators and the denominators are replaced with unknown integers that are randomly chosen at the time the problem is generated. The general form of this type of problem is:

$$\frac{a_1x+b_1}{c_1}-\frac{a_2x+b_2}{c_2}=\frac{a_3x+b_3}{c_3},$$

where a_i, b_i,and c_i are integers. Typically, these numbers are chosen within a range that will not make the problem too computationally unwieldy, though with the ubiquity of calculators, this can be relaxed. The following limitations are placed on these unknown integers to ensure exactly

one solution: (i) $c_1, c_2, c_3 \neq 0$ and (ii) $c_2c_3a_1 - c_1c_3a_2 - c_1c_2a_3 \neq 0$. After guaranteeing that a unique solution exists, we methodically generate the general solution to this problem template.

3.2 Generating Plausible Distractors

In problem solving, a *plausible* distractor would be one that corresponds to a specific, common error that a student can make when solving a problem or an observed student misconception. Plausible distractor solutions provide a way to evaluate specific content issues a student is having by consistently providing answer choices that correspond to common misconceptions. Moreover, they provide a more reliable assessment by avoiding the confounding of artificially similar answer choices. The process for creating plausible distractor solutions is nearly identical to the process for creating the correct solution, in that an exact, unique solution to a problem is found. The difference is that for distractor solutions, we construct *nearby problems* that are based on common errors students make when solving the original problem. Based on these errors, we can reverse engineer a problem, and then solve that problem algorithmically to obtain a *nearby solution* to the original problem. To make this more concrete, we present the creation of a distractor for the original problem.

A potential error that students may make when solving rational equations is that they do not divide each term in the numerator by the denominator. Essentially, students who do not have a complete understanding of rational expressions are solving the problem

$$\frac{a_1x}{c_1} + b_1 - \frac{a_2x}{c_2} - b_2 = \frac{a_3x}{c_3} + b_3.$$

In a similar way, distractors can be created for not dividing the first term in the numerator by the denominator or failing to distribute the minus sign in the numerator of the second term of the rational equation. These distractors are summarized in Figure 3 below.

Question 1. Solve the linear equation below.

$$\frac{-3x-6}{3}-\frac{-8x-8}{5}=\frac{7x+6}{2}$$

A. $x=\frac{-40}{29}$ *This corresponds to not distributing division throughout.*

B. $x=\frac{-34}{29}$ *This is the correct solution.*

C. $x=\frac{-66}{29}$ *This corresponds to not distributing division in the first term.*

D. $x=\frac{-17}{10}$ *This corresponds to failing to distribute the minus sign in the second term.*

Figure 3: The problem introduced in Figure 2 with the distractor solutions revealed and explained.

This method to automatically generate quality distractors can easily be extended to other observed issues that instructors see in students' work. Specifically, to generate a distractor from a known misunderstanding, solve the general template of the problem while committing the error(s) associated with the misunderstanding. Then, reverse engineer a nearby problem in the form of the original problem template, so that the nearby solution can be obtained in the same way as the original solution. This creates a plausible nearby solution that can be used as a distractor answer choice for the problem.

We have created plausible distractors that mirror common student errors made while completing an open-response version of this question. However, a student can find the correct solution to the previous example by taking each option and plugging it into the question, thereby rendering these distractors moot. The next section addresses this critical loophole in multiple-choice assessments by masking these distractors (and the solution) in intervals.

3.3 Disguising the Plausible Distractors and Solution

Solving algebraic problems presents a unique challenge for creating quality multiple-choice questions. When presented with a collection of options for the solution to a problem, students can test each of the potential solutions in the original equation and determine whether a given solution is valid. Considering this, additional measures must be taken to mask the answer choices to preserve the integrity of the distractors and ultimately generate quality assessments.

Conceptually, the additional layer for masking the answer choices is straightforward: replace the single-number answers with intervals that contain not only the corresponding single-number answer, but also infinitely many nearby numbers. This detaches students from the idea that they can test all the answer choices, because each answer choice contains an interval of infinitely many values that can be tested in the original problem. In Figure 4 below, we show an example of how the assessment question looks with the disguised answer choices.

Question 1. Solve the linear equation below. Then choose the interval that contains the solution.

$$\frac{-3x-6}{3}-\frac{-8x-8}{5}=\frac{7x+6}{2}$$

A. $x \in [-1.47, -1.21]$
B. $x \in [-1.21, -0.94]$
C. $x \in [-2.30, -2.10]$
D. $x \in [-1.78, -1.61]$

Figure 4: Multiple-choice example with masked solution and distractors.

Random, algorithmic interval generation itself is simple, compared to the method for generating distractor solutions described in the previous section. However, the problem-specific requirements for masking the answer choices can be a little more nuanced than the general algorithm for creating intervals. To create a quality disguise, it is necessary that the interval does not give clues as to the specific value that it is disguising. We do so by creating intervals that

must satisfy two criteria: (i) there is minimal overlap between intervals (as any overlap will not contain a solution) and (ii) the intervals do not reveal much information about the solutions they are disguising. We achieve this generation utilizing a normal standard distribution and interval checking using Python, but the interval generation need not be done in this way.

3.4 Method for Generating Multiple-Choice Items

We walked through how to generate a multiple-choice item based on a "Solve the equation" type question utilizing the 3-step model described by Gierl and Lai (2013). Here we explicitly describe how to include distractor generation into the 3-step model.

Step 1: Item Generation

In this step, the stem-type should be determined. This is equivalent to writing a free-response question and must include the stem and problem. In order to procedurally-generate versions of the question, elements of the stem and problem that can be modified must be identified at this point. A 1-layer model would be developed if only some small number of elements in the model can be modified. An n-layer model would be developed if many elements at multiple levels in a model can be modified.

Step 2: Content

In this step, the content knowledge required to solve the problem is determined. To accommodate the development of plausible distractors, any common errors or misconceptions associated to the problem should also be determined here. This can be collected by content specialists recalling common errors or misconceptions they are familiar with, recording any common errors identified in educational research experiments, or theoretically-predicted errors or misconceptions according to published mathematics education theoretical perspectives.

Step 3: Computer-based Algorithms

In this step, the content knowledge collected in step 2 is utilized to procedurally solve the problem. In addition, distractor solutions should also be generated by:

a) Isolating common conceptual misunderstandings or common errors related to the topic assessed by the problem.

b) Using these misunderstandings and/or errors to construct ``nearby problems".

c) Algorithmically solving these nearby problems to create a list of distractor solutions.

If the solution and distractor solutions are numeric in nature, the options can be disguised by algorithmically generating intervals that must satisfy two criteria:

a) There is minimal overlap between intervals (as any overlap will not contain a solution).

b) The intervals do not reveal much information about the solutions they are disguising.

To create distinct nearby problems based on common misconceptions or errors, the original stem/problem may need to be modified or a check may need to be created to regenerate the question until common misconceptions or errors do not produce the same solution as the correct solution.

4. Discussion of the Method

Now that we have demonstrated how multiple-choice items can be automatically generated, we can discuss the merits and limitations of our automatic item generation methods.

4.1 Merits

Efficient assessment generation - Distractor generation is simultaneously the most costly and critical component of writing multiple-choice assessments (Gierl, Bulut, Guo, & Zhang, 2017). In the literature, there were effectively three options when generating multiple-choice exams: (i) generate distractors based on similarity to the solutions (*weak theory*), (ii)

generate every distractor manually by relying on previous experiences with students or through experimental data (*strong theory*), or (iii) relying on education research that describes how students could develop their conception (Author & Author, 2019). While methods (ii) and (iii) are preferred to develop strong assessments, method (i) is commonly used due to the high costs of generating every distractor (Gierl & Lai, 2013). Our method generalizes and automates these distractors so that numerous items may be generated. In fact, some student errors (such as not distributing a negative) are so ubiquitous that they can be considered for a wide range of questions. This further reduces the time and effort a content specialist would need to generate distractors based on common student errors and misconceptions. Thus, our method for automatic item generation would allow for the cost-efficient development of numerous multiple-choice tests.

Multiple-Choice Assessment Integrity - One of the limitations of multiple-choice tests is the ability to assess students' procedural knowledge with integrity. This limitation is especially prevalent in K-14 mathematics, where questions will commonly require students to solve an equation (or system of equations) and provide possible solutions. A student needs only check these options in order until one satisfies the equation to arrive at the correct solution. To counter this limitation, we introduced a method to automatically generate intervals for each solution that effectively mask these options to prevent students from gaming the assessments. In unison with our distractor generation, we can automatically generate and mask multiple-choice options to assess students' procedural knowledge with integrity.

Formative assessment - Traditional multiple-choice assessments are used to determine whether students know or do not know some content. This is akin to knowing whether there is an issue with students' knowledge but does not effectively allow instructors to diagnose *why* there

may be an issue. By considering the distractors a student chooses over the course of one or more assessments, instructors can more accurately pinpoint *why* a student is not answering a question correctly. For example, during a multiple-choice assessment, a student may answer $5/20$ questions incorrectly. This student may have some minor issue with multiple content ideas, but it is also possible they are making the same common student errors (such as not distributing a negative correctly) over multiple questions. By tracking which solutions *and* distractors a student chooses throughout an entire assessment, we can more accurately assess if their issues are with the content or common mistakes. Moreover, this allows instructors to continuously evaluate foundational knowledge while simultaneously evaluating new content knowledge. These benefits illustrate that multiple-choice assessments *can potentially* provide diagnostic information, contrary to prevalent beliefs about multiple-choice assessments (Lissitz, Hou, & Slater, 2012).

Consequential merits from those described above include:

- **Potential for widespread use –** Unlike assessments developed by hand, these assessments can be used widespread once they are developed as they are efficient to generate and maintain their integrity even when the generation methods are shared.
- **Practical and Research Usefulness** – Assignments can be created for formative assessment in the classroom as well as for large-scale research use to test theoretical conception development.
- **Standardization of assessment** – Makes standardization of easy-to-generate assessments (e.g. aligned to State/National standards) possible.
- **Potential to use calculators –** By providing a method to disguise numeric-type options, the method allows for students to utilize calculators without dampening the integrity of the assessment.

4.2 Limitations

The method is not without limits. We discuss the most pressing issues with the method below, while also describing how these limitations can be mitigated.

High Start-Up Cost - Generating high-quality multiple-choice items normally requires a content specialist for distractor design. Our method would require either both a content specialist and someone with programming experience working side-by-side, or a content specialist with programming knowledge. For questions attempting to utilizing a theoretical perspective for how students with a misunderstanding or under-developed conception may answer, this would also require an education specialist. This further increases the start-up costs of developing multiple-choice assessments, making the method impractical for instructors with limited resources. However, once a series of items are created, they can be easily disseminated to other instructors. This task can be performed by those with the resources to do so and mass disseminated to other instructors.

Complication of Multiple-Choice Options - Masking the multiple-choice options, while effective in protecting the integrity of the assessment, does complicate students' choice of the solution. Rather than searching for the exact match of their answer, students would need to parse the interval notation language. Moreover, this may become confusing when the solution itself is an interval. For example, consider the inequality item in Figure 5.

Question 2. Solve the linear inequality below. Then, choose the constant and interval combination that describes the solution set.

$$8x - 6 > 10x \quad \text{or} \quad 5x - 5 < 8x$$

A. $(-\infty, a) \cup (b, \infty)$, where $a \in [1.5, 4.1]$ and $b \in [2, 4]$.
B. $(-\infty, a) \cup$, where $a \in [-9, -2]$ and $b \in [-8, 2]$.
C. $(-\infty, a) \cup$, where $a \in [-3, 5]$ and $b \in [-1, 5]$.
D. $(-\infty, a) \cup (b, \infty)$, where $a \in [-4.9, -1.6]$ and $b \in [-3, 0]$.
E. $(-\infty, \infty)$.

Figure 5: Automatically generated problem-solving systems of inequalities with interval answer choices.

While it may be second nature to instructors, students may struggle to interpret a phrase such as (a, ∞), where $a \in [a_1, a_2]$ for some a_1, a_2. This could lead to students solving the inequality correctly but choosing the wrong option. Addressing this limitation is a topic of future research.

5. Preliminary Results

Overall, our method is promising. It has been used to generate multiple exams for a large (800-1000 students annually), hybrid course of College Algebra. By leveraging Python, SageMath, and shell scripts written over the course of a year, complete exams and keys are generated *without any human input* in approximately 2.5 minutes. Two points to emphasize:

(1) *No technological skill is needed to create the exams at this point* (though the instructor may need assistance downloading the open-access software and files utilized by the authors) and

(2) *Exam generation would cost nothing to the instructors nor to the students.*

While data analysis for these assessments is ongoing, a summary of statistics for the Final Exam in Fall 2017 and Fall 2019 is provided below.

The Final Exam in Fall 2017 consisted of 25 multiple-choice questions with 4 options each. The majority of questions, 20/25, were taken from Pearson's College Algebra test bank while the other 5 were previous free-response questions (written by the instructor) and modified to be multiple-choice. The Final Exam in Fall 2019 consisted of 22 multiple-choice questions with 5 options each. These questions were generated using the procedure described in this paper. We analyzed three parts of each exam: (1) distractors, (2) how well individual items predicted student success, and (3) how consistent the exam was as a whole.

The procedure described in this paper for generating distractors provided ways to create *plausible* distractors – mistakes and misconceptions students could theoretically have. Literature

suggests an *effective* distractor is one that is chosen at least 5% of the time (Hingorjo & Jaleel, 2012). We could then consider *quality* distractors as those that students both theoretically could make (based on misconceptions or common errors) and do make during exams. We analyzed the distractors in both Final Exams in two ways: (1) categorizing the frequency of each individual distractor being selected (DS) some percentage of time and (2) calculating the number of items with *x* many distractors chosen at least 5% of the time. Tables 1-4 present a summary of these results. These percentages were done by version of the exam and then averaged for ease of discussion.

Error! Reference source not found. illustrates the percent of distractors that were selected by frequency. For example, Fall 2017 AVG 16% means that 16% of the distractors in Fall 2017 were not chosen by students for their respective question. Both exams had similar percentages of their distractors chosen through the exam. This suggests the novel procedure introduced in this paper was at least as effective as the non-computer-generated exam. This could also be considered a success for the computer-generated exam as it provided an additional distractor for each question and had the potential to provide an overabundance of theoretical distractors that students did not actually choose.

	Fall 2017				**Fall 2019**			
Distractor Selected (DS)	Ver A	Ver B	Ver C	**AVG**	Ver A	Ver B	Ver C	**AVG**
0%	9%	17%	23%	**16%**	19%	27%	15%	**20%**
0% < DS < 5%	33%	29%	40%	**34%**	38%	27%	33%	**33%**
5% < DS < 10%	29%	29%	17%	**25%**	20%	28%	30%	**26%**
10% < DS < 15%	16%	11%	11%	**12%**	15%	7%	10%	**11%**

15% < DS < 20%	3%	4%	1%	**3%**	3%	5%	7%	**5%**
DS > 20%	8%	9%	8%	**8%**	5%	6%	5%	**5%**

Table 1: Percentage of distractors selected by students out of total number of distractors in Fall 2017 and Fall 2019.

Error! Reference source not found. illustrates a by-question analysis of the distractors by considering the number of items with *x* many distractors chosen at least 5% of the time. Again, we note that the computer-generated exams provided 4 distractors, while the non-computer-generated exam had only 3 distractors. Here we see clear advantages to the computer-generated distractors. It averaged generating 5% of the exam items with *all 4 distractors* chosen by students and an additional 15% average of exam items with 3 effective distractors. The largest difference was in the number of questions with no effective distractors: 9% average for the computer-generated exam versus 21% average for the non-computer-generated exam. This is a clear success of the distractor generation method – it provided at least one quality distractor for a large majority of the exam (91%).

	Fall 2017				**Fall 2019**			
Items with x distractors chosen >5%	Ver A	Ver B	Ver C	**AVG**	Ver A	Ver B	Ver C	**AVG**
4	NA	NA	NA	**NA**	0%	9%	5%	**5%**
3	28%	24%	8%	**20%**	18%	14%	14%	**15%**
2	28%	32%	28%	**29%**	45%	32%	27%	**35%**
1	32%	24%	32%	**29%**	27%	41%	41%	**36%**
0	12%	20%	32%	**21%**	9%	5%	14%	**9%**

Table 2: Percent of questions with x distractors chosen by more than 5% of students.

In Item Response Theory, statistics are used to measure the relationship between performance on individual assessment items and the overall assessment (Varma, 2006). The Point-Biserial Correlation (PBC) is a common correlation measure for assessments, where a positive PBC corresponds to a high-achieving student marking the question correctly while low-achieving students marking the question incorrectly. A PBC of 0.1 or higher is considered desirable while simultaneously avoiding negative PBCs, which are indicative of low-achieving students marking correctly what high-achieving students mark incorrectly (Varma, 2006). **Error! Reference source not found.** categorizes the percentage of questions that fall within the identified ranges. First, it should be noted that both exams have a large percentage of predictive questions: 88% and 91% respectively. They both also have similar numbers of problematic questions (1% and 2%) and suspect questions (5% and 8%). Like the Distractor Selection analysis, this suggests the computer-generated exam is at least as effective at generating quality distractors as the non-computer-generated exam.

	Fall 2017				**Fall 2019**			
Point Biserial Correlation	Ver A	Ver B	Ver C	**AVG**	Ver A	Ver B	Ver C	**AVG**
PBC < 0	0%	0%	4%	**1%**	0%	0%	5%	**2%**
0 < PBC < 0.15	8%	0%	8%	**5%**	5%	14%	5%	**8%**
0.15 < PBC < 0.25	4%	12%	16%	**11%**	0%	14%	18%	**11%**
PBC > 0.25	84%	80%	68%	**77%**	95%	73%	73%	**80%**

Table 3: Percentage of assessment items in point biserial coefficient ranges.

Finally, the KR-20 reliability coefficient is used to estimate the internal consistency reliability of an assessment (Salvucci, Walter, Conley, Fink, & Saba, 1997). In other words, the

reliability coefficient attempts to measure whether another group of similar students achieve in a similar way. Salvucci, Walter, Conley, Fink, & Saba (1997) proposed the following interpretations of KR-20 coefficients:

- Less than 0.5, the reliability is low;
- Between 0.5 and 0.8, the reliability is moderate;
- Greater than 0.8, the reliability is high (p. 115).

Error! Reference source not found. illustrates that both exams are in the upper-moderate range. As with much of the other data, this suggests the computer-generated exams are at least as good as the non-computer-generated exams. However, this is another clear win for the computer-generated exams as this result illustrates that the code effectively controlled for changing the individual items *without changing their difficulty,* a potential issue described when detailing the procedure.

	Fall 2017				**Fall 2019**			
	Ver A	Ver B	Ver C	**AVG**	Ver A	Ver B	Ver C	**AVG**
KR20	0.80	0.79	0.60	**0.73**	0.73	0.65	0.69	**0.69**

Table 4: KR-20 correlation coefficients.

At each level of the data analysis (distractor, item, and overall exam), the computer-generated exam was shown to be at least as effective as the exam created by Pearson and the instructor. The one major difference was in the number of items with at least one quality distractor: 91% versus 79%. Combining this with the clear advantages in amount of time to create an exam (2.5 minutes for all 3 versions of Fall 2019) and dynamic nature of the computer-generated exams, the procedure appears to be effective at generating quality distractors and quality multiple-choice exams in general.

6. Conclusions

Automated item generation is not a novel concept in the assessment literature and has been discussed as early at 1969 (Bormuth, 1969). Since then, copious guidelines for developing multiple-choice items have been developed and agree that distractors play a fundamental role in multiple-choice tests. For example, Gierl et al. (2017) consider distractors to (i) require a significant amount of time and resources to create, (ii) affect item quality and learning outcomes, and (iii) provide diagnostic inferences about students' test performance. The authors go on to say that "distractor development, in fact, is often considered by content specialists to be the most daunting and challenging component of writing a multiple-choice item" (p. 1086). Yet, automated distractor generation has received relatively little attention, even in the context of mathematics (Gierl, Lai, Hogan, & Matovinovic, 2015). When automatic distractor generation has been explored, it has largely been relegated to manipulating the solution of an item in some minor way or by mapping all possible solutions to the (relatively simple) structure of an item (Gierl, Bulut, Guo, & Zhang, 2017).

Note that distractor generation is distinct from the approach Intelligent Tutoring Systems (such as ALEKS) take that utilize student *Knowledge Spaces* – a pair (Q, K) consisting of some set of Questions (Q) and a subset of questions (K) that represent the questions a student could answer correctly (Cosyn & Thiery, 2000). At a fundamental level, Knowledge Spaces operate by identifying the questions a student can and cannot complete in some progression to determine the next question their knowledge would allow them to start on. For a simplistic example, consider a concept to have 6 linear questions that build up to a robust understanding. The system would start by asking the student to complete Q1 – if the student is correct, it could move on to Q2 or beyond. If the student was correct with Q1, it moved to Q3, and the student was then incorrect,

the student would have the knowledge space K = {Q1} and thus be taught the knowledge needed to answer Q2. While an oversimplification of Knowledge Spaces, this example illustrates that Intelligent Tutoring Systems work through correct/incorrect and not *theoretically why* a student is incorrect. Moreover, these systems purposely avoid multiple-choice items to further correlate a correct answer to sufficient knowledge to answer the question. Thus, our work on distractor generation is fundamentally different than the computer-generated questions Intelligent Tutoring Systems employ.

We presented a novel method for dynamically generating distractors by creating *nearby problems* that can be algorithmically solved via computers. For a given concept, the instructor decides what content will be evaluated, and chooses the corresponding stem template, from which the problem is algorithmically generated. This formal statement of the problem can be procedurally solved to find the correct solution. Then, using common student misconceptions, *nearby problems* can be constructed and then solved using the same procedure that solved the original problem. These *nearby solutions* are plausible distractor solutions corresponding to specific content areas with which students struggle.

Moreover, we introduced a method of masking these solutions and distractors to prevent students from working backwards from answer choices the correct solution, thus preserving the integrity of these automatically generated multiple-choice items. For problems with single-number answers, we propose hiding the solutions (distractor or otherwise) within non-overlapping intervals that contain not only the corresponding single-number answer, but also infinitely many nearby numbers. This detaches students from the idea that they can test all possible choices, because each answer choice contains an interval of infinitely many values that can be tested in the original problem. However, students who obtained a solution (distractor or

otherwise) will be able to easily identify the appropriate answer choice. Numerous methods for generating these intervals can be effective, as long as there is minimal overlap in the intervals and the intervals do not reveal information about the option it is disguising.

Dynamically generating distractors associated with student misconceptions and errors holds a variety of theoretical merits. First and foremost, it allows for the cost-efficient development of numerous multiple-choice assessments. Constructing a single multiple-choice, n-layer item can result in hundreds (or even thousands) of questions with relevant distractors. In unison with our method to mask options, multiple-choice assessments can be efficiently used to assess students' procedural knowledge with integrity. Dissemination of these automatically generated assessments can help solve a practicality issue with educational research (Van Velzen, 2013) by bridging the gap between the research and practice. Theoretically speaking, as generated distractors are associated to student misconceptions and errors (rather than small perturbations of the correct solution), these assessments can be used to help diagnose *why* a student did not answer a question correctly and could counter the misconception that multiple-choice assessments cannot provide diagnostic information (Lissitz, Hou, & Slater, 2012). This method also allows instructors to track misconceptions and small errors through multiple assignments, allowing for the continuous evaluation of foundational knowledge while simultaneously evaluating new content knowledge. Tracking misconceptions and small errors could potentially lead to partial credit on multiple-choice items and create free-response-like grading. It can also allow the development of semester-long feedback systems that track student development of critical concepts.

Our method is not without limitations. It further increases the start-up costs of developing multiple-choice assessments, making the method impractical for instructors with limited

resources. This, however, can be mitigated by the generality of the method. In addition, masking the distractors and solutions complicates the option decision process, which may lead to students solving the problem correctly but choosing the wrong option.

7. Conflicts of Interest

No potential conflict of interest is reported by the authors.

References

Bormuth, J. (1969). *On a theory of achievement test items.* Chicago, Illinois: University of Chicago Press. doi:10.1086/443056

Briggs, D. C., Alonzo, A. C., Schwab, C., & Wilson, M. (2006). Diagnostic assessment with ordered multiple-choice items. *Educational Assessment, 11*, 33-63. doi:10.1207/s15326977ea1101_2

Carlson, M., Jacobs, S., Coe, E., Larsen, S., & Hsu, E. (2002). Applying covariational reasoning while modeling dynamic events: A framework and a study. *Journal for Research in Mathematics Education, 33*(5), 352-378.

Carlson, M., Oehrtman, M., & Engelke, N. (2010). The precalculus concept assessment: A tool for assessing students' reasoning abilities and understandings. *Cognition and Instruction, 28*(2), 113-145. doi:10.1080/07370001003676587

Cauley, K., & McMillan, J. (2010). Formative assessment techniques to support student motivation and achievement. *The Clearing House: A Journal of Educational Strategies, Issues and Ideas, 81*(1), 1-6. doi:10.1080/00098650903267784

Collins, J. (2006). Writing multiple-choice questions for continuing medical education activities and self-assessment modules. *Radiographics, 26*, 543-551. doi:10.1148/rg.262055145

Cosyn, E., & Thiery, N. (2000). A practical procedure to build a knowledge structure. *Journal of Mathematical Psychology, 44*(3), 383-407. doi:10.1006/jmps.1998.1252

Filloy, E., & Rojano, T. (1989). Solving equations: The transition from arithmetic to algebra. *For the Learning of Mathematics, 9*(2), 19-25. Retrieved August 16, 2018, from https://www.jstor.org/stable/40247950

Frey, B. B., Petersen, S., Edwards, L. M., Pedrotti, J. T., & Peyton, V. (2005). Item-writing rules: Collective wisdom. *Teaching and Teacher Education, 21*, 357-364. doi:10.1016/j.tate.2005.01.008

Gierl, M. J., & Lai, H. (2013). Using automated processes to generate test items. *Educational Measurement: Issues and Practice, 32*, 36-50. doi:10.1111/emip.12018

Gierl, M. J., Bulut, O., Guo, Q., & Zhang, X. (2017). Developing, analyzing, and using distractors for multiple-choice tests in education: A comprehensive review. *Review of Educational Research, 87*(6), 1082-1116. doi:10.3102/0034654317726529

Gierl, M. J., Lai, H., Hogan, J., & Matovinovic, D. (2015). A method for generating test items that are aligned to the common core state standards. *Journal of Applied Testing Technology, 16*, 1-18. Retrieved August 16, 2018, from http://www.jattjournal.com/index.php/atp/article/view/80234

Haladyna, T. M., & Rodriguez, M. C. (2013). *Developing and validating test items.* New York, NY: Routledge.

Hingorjo, M. R., & Jaleel, F. (2012). Analysis of one-best MCQs: The difficulty index, discrimination index and distractor efficiency. *JPMA-Journal of the Pakistan Medical Association, 62*(2), 142-147.

Leatham, K. R. (2014). *Vital directions for mathematics education research.* Springer Science & Business Media.

Lissitz, R., Hou, X., & Slater, S. (2012). The contribution of constructed response items to large scale assessment: Measuring and understanding their impact. *Journal of Applied Testing Technology, 13*(3), 1-50.

Moreno, R., Martinez, R. J., & Muniz, J. (2015). Guidelines based on validity criteria for the development of multiple choice items. *Psicothema, 27*, 388-394. doi:10.7334/psicothema2015.110

Rodriguez, M. C. (2011). Item-writing practice and evidence. In S. N. Elliott, R. J. Kettler, P. A. Beddrow, & A. Kurz (Eds.), *Handbook of accessible achievement tests for all student: Bridging the gaps between research, practice, and policy* (pp. 201-206). New York, NY: Springer.

Salvucci, S., Walter, E., Conley, V., Fink, S., & Saba, M. (1997). *Measurement error studies at the National Center for Education Statistics.*

Van Velzen, J. (2013, August). Educational researchers and practicality. *American Educational Research Journal, 50*(4), 789-811. doi:10.3102/0002831212468787

Varma, S. (2006). *Preliminary item statistics using point-biseral correlation and p-values.* Morgan Hill CA: Educational Data Systems Inc.