The Future of Engineering Education
2024 Annual Conference & Exposition
Oregon Convention Center
Portland, OR . June 23 - 26, 2024
ASEE
Paper ID #41262

Keys to Success for an Alternative Grading Scheme in a Large Enrollment Differential Equations Course

Dr. Hadas Ritz, Cornell University

Hadas Ritz is a senior lecturer in Mechanical and Aerospace Engineering, and a Faculty Teaching Fellow at the James McCormick Family Teaching Excellence Institute (MTEI) at Cornell University, where she received her PhD in Mechanical Engineering in 2008. Among other teaching awards, she received the 2021 ASEE National Outstanding Teaching Award.

Stephan Wagner, Cornell University

Stephan Wagner is a senior undergraduate student in Mechanical Engineering with a professional background in consumer software and technology.

Work-in-Progress: Keys to Success for an Alternative Grading Scheme in a Large Enrollment Differential Equations Course

Abstract

In this work-in-progress paper, we describe the key changes we made to our efforts to implement a standards-based, mastery-based grading scheme in a large enrollment Differential Equations course at an R1 university. In this continuation of prior work, we have addressed many of the challenges we faced in our first effort to transition from traditional grading to an alternative grading scheme in this course.

In the first version of our implementation of an alternative grading scheme, we identified inconsistency in grading as a major shortcoming. In this paper, we will describe the measures we instituted to improve grading fidelity. Other significant shortcomings in the first version were a consequence of scale: alternative grading schemes in large enrollment courses (in this case, 400+ students) bring unique challenges. The steps we have taken include improving our own pre-semester preparedness and increasing automation of all logistics.

We will describe custom Python and MATLAB software written to address each aspect of the logistics that proved a major obstacle in the past. These tools are specific to the learning platforms in use at our school: Canvas as the Learning Management Software (LMS) and Gradescope for grading most hand-written work. Our "lessons learned" from two versions of this alternative grading scheme are presented here as "best practices" which we hope will be useful for other faculty wishing to implement standards-based grading on a large scale.

Introduction

Alternative grading schemes encompass a large variety of course assessment rubrics and a large variety of implementations of the different styles. Some examples include Mastery Based Grading, Standards Based Grading, Specifications Grading, and Ungrading, among others [1, 2]. Motivations for implementing a course assessment scheme different from a traditional point-based rubric include encouraging a growth mindset in students, reducing testing anxiety which may occur due to high-stakes exams, and requiring students to solve problems correctly instead of awarding partial credit for incorrect work. While these grading approaches have been used successfully in a wide variety of courses, implementation in courses of the scale we describe here is far from a solved problem. Even the phrase "large enrollment" may mean very different things to different instructors. A class size of 76 is sometimes described as large[3] but still allows for reassessment during office hours as a foundational aspect of the course design. That could not work with over 400 students.

In prior work [4], we described our first effort to implement a standards-based, mastery-based grading scheme in a large enrollment Differential Equations (for Engineers) course and identified the following significant barriers to a fully successful implementation:

- inconsistency in grading,
- unacceptably delayed availability of assessment opportunities,
- lack of automation for several tasks related to grades and communication.

In this paper, we provide a brief description of the course, describe how we addressed each of the items above, and summarize the remaining areas of needed improvement moving forward.

Course Structure

This 400+ student, 4 credit hour course covers ordinary and partial differential equations at the sophomore level. Students attend three 50-minute lectures and one 50-minute TA-led discussion section each week. In the grading scheme we have implemented, we identified 21 content-based Learning Outcomes (LOs), which are listed in Table 1. This list of LOs is a slight modification of the first version; it puts a greater emphasis on requiring students to identify the correct technique to apply instead of being instructed on the nature of the problem or which technique to employ in solving it, and a greater emphasis on open-ended modeling problems as a significant learning objective of the course.

For each LO, students need to demonstrate mastery one or two times (depending on the LO), which they do by solving a problem entirely correctly on an in-person quiz or exam, or submitting an entirely correct online Checkpoint quiz. Table 2 shows the grading scale employed in evaluating student mastery. For a solution that is almost entirely correct but with a minor non-conceptual error, students are able to revise with sufficient reflection and convert the score to a successful demonstration of mastery. Because of the strict grading of individual problems, multiple opportunities (two to five) must be available for most LOs, except those covered towards the very end of the semester.

The Checkpoints are Canvas quizzes—partially auto-graded, partially manually graded—taken and submitted by students outside of class in an unproctored environment. To help maintain academic integrity, we needed large banks of randomized questions. Building these Checkpoint quizzes in a way that allows randomization but relatively efficient grading is a crucial part of a successful implementation of our grading scheme. Final course grades are based entirely on the number of LOs mastered; homework is assigned but not collected or graded, serving strictly as practice opportunities for the students, with full solutions available with each assignment.

The main aspects of this course structure that proved problematic in our initial implementation were inconsistency in applying the Check/Almost/Not Yet scoring, and lack of timely availability of Checkpoints. The following sections go into more detail about how we addressed these issues.

Table 1: Learning outcomes for the course. Students had to demonstrate mastery of starred (⋆) LOs twice to earn credit towards their final course grade, and other LOs a single time.

LO	Description
⋆Direction fields	Identify the direction field associated with a first order differential equation and sketch a particular solution or a representative family of solutions.
⋆Separable	Determine whether a first order ordinary differential equation is separable. Solve a first order separable differential equation using integration.
⋆Integrating factor	Solve a first order linear differential equation by using an integrating factor.
Existence and Uniqueness	Determine whether a first order ODE has a solution and, if so, whether it is unique. Recognize additional solutions for differential equations with distinct solutions.
⋆Autonomous ODEs	Determine stable and unstable equilibria of autonomous ordinary differential equations. Sketch representative solutions with an emphasis on behaviors near equilibria.
Initial Value Problems	Given a first order differential equation, choose an appropriate solution method, determine the general solution, and determine particular solutions satisfying appropriate ICs.
⋆Numerical I	Solve a first order ODE using Euler's method, graphically and algebraically. Identify error and stability. Describe differences between the implicit and explicit methods.
Modeling I	Recognize situations in which a first order differential equation is relevant. Develop an appropriate mathematical model of such systems, choose an appropriate technique for analyzing or solving the problem, and carry out the analysis.
⋆2nd order ODEs	Given a homogeneous second order linear differential equation with constant coefficients, find the general solution.
Reduction of Order	Given a linear ordinary differential equation of second or higher order, use the reduction of order technique to find the general solution.
Undetermined coefficients	Determine a particular solution to a nonhomogeneous second order linear differential equation with constant coefficients using the technique of undetermined coefficients.
Vibrations	Determine whether a second order differential equation represents a free or forced system with or without damping. Identify whether resonance will occur. Find the solution and analyze the results in context.
Higher Order ODEs	Given a higher order linear ODE with constant coefficients, find the general solution.
Numerical II	Numerically solve a higher order differential equation or system of first order differential equations using explicit Euler integration.
Modeling II	Recognize situations in which a first order or higher order ordinary differential equation is relevant. Develop a mathematical model of such systems, choose an appropriate technique for analyzing or solving the problem, and carry out the analysis.
BVPs	Determine the eigenvalues and eigenfunctions associated to an ordinary differential equation with boundary conditions. Identify all solutions to a given boundary value problem.
Fourier series	Calculate the Fourier series associated with a function. Determine convergence.
Even/Odd Fourier Series	Determine whether a Fourier sine or cosine series is appropriate for a model and calculate it. Determine convergence of the series.
Heat Equation	Determine the general solution to the homogeneous heat equation and analyze the results in context.
Nonhomo. Heat	Sketch solutions to the heat equation based on non-homogeneous BCs.
Homogeneous Differential Equations	Given the equation for a homogeneous differential equation, determine the solution satisfying appropriate initial and/or boundary conditions. Note that this can encompass ODEs, the heat equation, the wave equation, and the Laplace equation.
Nonhomogeneous Differential Equations	Given the equation for a nonhomogeneous differential equation, determine the solution satisfying appropriate initial and/or boundary conditions. Note that this can encompass ODEs, the heat equation, the wave equation, and the Laplace equation.

Table 2: Possible scores on each problem in this mastery grading scheme. If a successful revision is submitted, a 2 can be upgraded to a 3.

Score	Description
3: Check received	This indicates a complete and correct solution up to and including the final answer. This is one "mastery check" on the related LO.
2: Almost	Non-conceptual error such as an algebra error or misread number in the given problem, etc. Students can revise their solution and resubmit it.
1: Not yet	This indicates a student tried to solve the problem, but did not demonstrate understanding of the problem or solution methodology. A conceptual error exists. They have to try again on a new problem at the next opportunity.
0: Not attempted	This is just a flag to say that a student didn't try to answer a problem. If students are missing quizzes we want an easy way to notice so we can reach out and see what's going on.

Grading

Consistently applying the grade of Check/Almost/Not Yet is particularly important because individual problem scores completely determine course grades. To ensure such consistency, the two instructors and seven TAs met weekly to discuss grading of each quiz problem. We looked at a selection of student work to get a sense of the types of mistakes made, wrote rubric items specific to each type of error, and categorized each error into a Revise/Almost (2) or Reattempt/Not Yet (1) designation. We reached consensus as a group, sometimes resorting to a vote on how to treat each kind of error. One lesson learned in our prior work was that when "revise" grades were handed out too leniently, students didn't take their first attempt at an assessment seriously enough. Holding in-depth conversations about every quiz problem significantly improved grading consistency and fairness. It also gave the instructors the opportunity to remind the TAs about requiring sufficient responses to the required reflection questions for any revisions submitted. The learning benefit of the revision process comes from requiring the students to engage in self-reflection about their errors, why they were wrong, and how to avoid them in the future.

Custom Software

Before the semester began, we developed a suite of software tools to automate certain course tasks[5].

The most significant problem with the first version of our alternative grading implementation was a lack of timely access to assessment opportunities. Instead of having everything prepared before the semester began, we naively thought we would be able to generate the necessary Canvas Checkpoint quizzes in the midst of the semester. Unfortunately, the demands of such a large enrollment course (along with other obligations) meant the Checkpoint quizzes were not available until close to the end of the course. To address this challenge, we built a MATLAB-based generator for Canvas checkpoints. Of the many types of questions offered in Canvas quizzes, the generator supports the following: matching, multiple answers, multiple blanks, multiple choice, multiple dropdowns, and open-ended numeric. Questions of these types can be added to the generator in any combination and given a respective set of correct and incorrect answers. Each question's answers, as well as the questions themselves, can be shuffled. The generator will then upload the complete checkpoint to Canvas as a quiz. This tool allowed us to generate large banks of similar questions with some randomization. We were able to generate all necessary Checkpoint opportunities before the semester began, deploying them in a timely manner throughout the course, preventing a logjam of testing at the end of the semester.

The final major obstacle to success in our first alternative grading implementation was the lack of automation of certain tasks such as: handling revisions of quizzes, exams, and Checkpoints; making Checkpoint opportunities available only to those students who needed them; giving students with testing accommodations extended time on the timed Checkpoints; and communicating course progress throughout the semester. We addressed each of these areas with custom software.

Our conditional assignment system applies both to revision opportunities and Checkpoints. For revision opportunities, the software identifies students who earned a score of Almost/Revise on a quiz or exam problem, and creates a Canvas assignment where those students can submit a revision. Students who do not qualify for a revision do not see the assignment. This makes it clear to both students and course staff when such revisions are due. The revision format allows graders to easily see students' original work next to their revision, making grading more efficient. For Checkpoints, the software opens Checkpoints generated in MATLAB only to students who need the mastery opportunity. The tool scans the Canvas learning mastery gradebook, identifies students who have yet to complete a given LO, and assigns a given Checkpoint to them.

Handling accommodations appropriately is an obligation of course staff and applies to timed Canvas Checkpoint quizzes as well as in-person quizzes and exams. With over 400 students in the course, we could easily have up to 50 students needing extended time on timed assessments. Unfortunately, at the time of the course, Canvas did not have an automatic system to apply accommodations to quizzes. Our software takes in a record, downloaded from our university's Student Disability Services faculty portal, that includes information about all accommodations for students in our course. The tool then automatically applies the appropriate amount of extra time (based on the base time for the particular quiz and the percentage extra time for every individual's accommodations) to each student. This ensured privacy of students' accommodation status while meeting each student's needs.

On Canvas, completion of LOs is tracked by a rubric system, but the rubrics are not scored automatically when students complete Canvas quizzes. We therefore developed an autograder tool that scans a given quiz for submissions, transforms scores to their learning mastery equivalents, and updates the rubrics accordingly.

Other mastery opportunities, such as discussion quizzes and exams, were managed and graded through Gradescope, but that service does not support syncing to a learning mastery gradebook on Canvas. We built a gradebook synchronizer tool that takes in point-based grade data exported from Gradescope, transforms it into learning mastery scores, and uploads these to Canvas.

Despite the existence of a learning mastery gradebook on Canvas, learning outcome grade data could not be exported from the platform at the time of the course. Our last bit of software exports the learning mastery gradebook as a single file holding rows for students and columns for the status of each LO. This allowed us to perform a variety of desired analysis, such as generating plots of the number of opportunities students were skipping, how many students had completed each LO or a certain number of LOs, etc. That was extremely helpful to communicate not just individual progress to students, but peer-relative course standing and whether they were making appropriate progress throughout the semester, as well.

Conclusion

Implementing novel pedagogy or grading schemes in a large enrollment course presents special challenges and requires thoughtful preparation. Any inefficiency or error in any aspect of the course logistics gets magnified many times over. After a first attempt that uncovered specific failure modes, we addressed them with course staff training, adequate pre-semester preparation, and an invaluable suite of custom software tools built to meet our specific needs. Several aspects of the software may be useful to a broader community of faculty implementing alternative grading schemes with the learning platforms we use (Canvas and Gradescope), particularly in cases such as accessing the complete Canvas learning mastery gradebook, assigning Checkpoints only to those students who need them, and automating extended time accommodations on Canvas quizzes.

While many of the logistical hurdles have been solved, the overall transition to a successful mastery-based and standards-based grading scheme remains challenging. Student resistance to the unfamiliar and exacting grading scheme has been a problem. Without historical data of how progress throughout the semester translates to end-of-term course grades, student anxiety was higher than we expected and hoped. From end-of-semester course evaluations, we noted that several students shared a perception that the grading scheme favors those who succeed early in the semester, and that it is particularly difficult to recover from a bad start. Considering the grading scheme is partially intended to avoid just such an outcome, it is unclear whether this perception is accurate, though the mere perception can be detrimental to student buy-in. Continued work is necessary to refine this system into one that best supports student learning in an efficient and scaleable manner.

References

[1] D. Clark and R. Talbert, *Grading for growth: A guide to alternative grading practices that promote authentic learning and student engagement in higher education*. Taylor & Francis, 2023.

[2] L. B. Nilson and C. J. Stanny, *Specifications grading: Restoring rigor, motivating students, and saving faculty time*. Routledge, 2015.

[3] J. Elsinger and D. Lewis, “Applying a standards-based grading framework across lower level mathematics courses,” *PRIMUS*, vol. 30, no. 8-10, pp. 885–907, 2020.

[4] H. Ritz, “WIP: Implementing an alternative grading scheme in a large enrollment differential equations course: Lessons learned,” in *2023 ASEE Annual Conference & Exposition*, 2023.

[5] S. Wagner and H. Ritz, “Canvas mastery toolkit,” 2024.