Overcoming non-numerical challenges in an engineering numerical methods course

Dr. Ivan Detchev, University of Calgary

Ivan Detchev holds a BScE (first division) from the department of Geodesy and Geomatics Engineering at the University of New Brunswick. He also obtained a MSc and a PhD in Geomatics Engineering from the University of Calgary. Dr. Detchev is currently an instructor in surveying and mapping at the University of Calgary. He is interested in the scholarship of teaching and learning (SoTL) related to engineering education.

Dr. Elena Rangelova, University of Calgary

Dr. Elena Rangelova is a senior instructor in the Department of Geomatics Engineering, Schulich School of Engineering, University of Calgary. She received her PhD degree from the same department in 2007. Her research interests in scholarship of teaching and learning are in the field of deep, active and team-based learning, as well as transformative learning in threshold concepts.

Ms. Sheng Lun (Christine) Cao, University of Calgary

Sheng Lun (Christine) Cao is a second-year Master of science student at the Schulich School of Engineering, University of Calgary. Her primary research field is in applied machine learning on urban planning and development. Due to her interest in Engineering Education, Christine also works as a research assistant for Dr. Elena Rangelova and Dr. Ivan Detchev.

Overcoming non-numerical challenges in an engineering numerical methods course

Abstract

This paper addresses the application of some of the current pedagogical practices in an engineering numerical methods course. The paper describes the course and explains its challenges. It then briefly goes over the theoretical framework and the engineering accreditation requirements which shape its design and development. The course design, its implementation, and observations performed by a third-party research assistant are listed next. In particular, instructional remedies developed in order to improve students' learning experience are detailed. Lastly, the course instructor and the research assistant discussed some of the improvements and unforeseen student behaviour. Note that the course instructor is a new engineering educator who would like to share his course design, get feedback on the implemented course developments, and in general use this as an opportunity to self-reflect on the changes made to the course and how they can be scaled for other offerings of the course in the future.

Introduction

This paper is about a numerical methods course in an engineering faculty at a Canadian university. This is a common-core engineering course taken by primarily civil, mechanical, and geomatics engineering students in either their second or third year. The topics in the course include numerical error, solutions to linear and non-linear equations and systems of equations, interpolation, curve fitting / least squares estimation, numerical differentiation and integration, and ordinary differential equations (both initial and boundary value problems). The course content is based off of a standard textbook in numerical methods which uses MATLAB as a programming language. The learning outcomes include choosing appropriate methods for solving a given engineering problem and proficiently programming the algorithms associated with the methods in MATLAB. The course normally runs as three hours of lectures (two sections) and two hours of tutorials (again, two sections) a week. The assessment items are typically five assignments, one or two midterms, and a final exam. Usually there are 100-200 students enrolled in each section.

This course faces several challenges. One of the major issues is that the student demographic can be divided into two distinct groups – students who have no MATLAB background, and students who have experience programming in MATLAB in at least one course prior to enrolling in numerical methods. Amongst the student population, the course has the stigma of being a dry math course. Since it is a terminal course, many of the students lack the motivation of performing well in it other than for getting the grade and credit for the course. In addition, there have been two attempts at teaching the course in a blended manner (i.e., an online offering with limited in-person contact hours) with mixed reviews. The latest developments in this course have been remedies addressing the above-listed challenges. A pilot iteration of the course was run in a fundamentally different way. This was done during the spring intersession of 2019, where there were 65 students enrolled in the course.

The next two sections describe the theoretical framework and the engineering accreditation requirements which shaped the new course design and developments. The design, the implementations, and work done by a third-party research assistant are explained in the methodology section. The paper ends with a discussion on what seemed to work, and what could realistically be scaled up to a larger class-size in a future course offering.

Theoretical framework

The numerical methods course in question contributes to the accreditation of a number of engineering programs at the school and as such it must abide by the requirements set by the engineering accreditation body in Canada. It is however worth explaining a broader theoretical framework before delving into how the accreditation constraints influence the design of the course. This is addressed next in terms of both the course design / development and the course delivery.

In terms of course development, the theoretical framework that is advised to be used in the design of engineering courses is Bloom's taxonomy [1], and more specifically the version of the taxonomy modified by Krathwohl [2]. Bloom's taxonomy presents a congnitive spectrum or a hierarchy used for the classification of learning tasks (see Table 1). At the lowest level of the hierarchy is ”remembering”, i.e., where only memorizing facts is required, while at the highest level is ”creating”, i.e., where a great deal of critical thinking is necessary. In addition, action verbs depicting the learning tasks can be grouped under a specific level in the hierarchy (see Table 1). When expressing the learning outcomes for an entire course (e.g., in the course outline), for a course module, for homework or laboratory assignments, or for lesson planning in general, it is recommended to use such action verbs. If possible and/or applicable the aligned course components (i.e., the learning outcomes, the learning and teaching exercises in support of the learning outcomes, and the graded items assessing the learning outcomes) should cover the full spectrum of cognitive levels for the course to be considered as well-rounded. For example, if the course material contains mostly definitions, basic concepts, and problems requiring a one-step solution, students are most likely not challenged at the higher congnitive levels, and it may be wise to introduce outcomes that provoke more complex probem solving and critical thinking. On the other hand, if students are expected to analyse complex results, evaluate multiple routes to a complicated solution, generate a product of value, or even contribute to the body of knowledge in the discipline, the instructor would have to scaffold the learning process with more rudimentary tasks at the beginning, and have the higher level tasks culminate towards the end. Note that the action verbs listed in Table 1 are meant as examples only. It is possible that depending on the context certain action verbs may be catagorised under different cognitive levels.

In terms of a course delivery, the theoretical framework is based on active learning [3]. According to Prince [3] active learning includes learning and teaching activities where students are intellectually engaged beyond passively listening to a lecturer and/or mechanically copying notes. Example active learning activities may include answering review questions via clickers, writing reflective minute papers on muddiest concepts, think-pair-share, etc. Generally, the more active learning is incorporated in a course the better are the long term benefits for students such as retention of learned material [4]. Of course this should be gauged as per the type of course, the specific group of students and the instructor's teaching philosophy. A form of active learning, the main author is currently experimenting with, is inspired by team-based learning (TBL) [5]. Team-based learning most definitely belongs to the family of active learning activities. In TBL students need to prepare for class by reading certain material, then in class they first write an individual quiz, and immediately after the individual quiz they tackle the same quiz in small groups. In a later class, after both the individual and group quizzes have been graded, the instructor must address any misconceptions via a mini lecture. The TBL cycle on a particular lesson or course module culminates in an application, i.e., the students are now assigned a more complex problem to solve in the same small groups.

Table 1. Examples of action verbs appropriate for engineering for each level in the cognitive hierarchy of Bloom's taxonomy

Order in hierarchy	Cognitive level	Example action verbs
6.	Create	Design, develop, modify, generate, invent
5.	Evaluate	Check, interpret, criticize, decide, refine/revise
4.	Analyse	Identify, differentiate, select, correlate, conclude
3.	Apply	Use, compute, solve, implement, demonstrate
2.	Comprehend	Summarize, classify, compare, contrast, discuss
1.	Remember	Recall, list, define, describe, explain

Engineering accreditation requirements

In Canada, there is a nation-wide organization for the accreditation of engineering programs. It is called the Canadian Engineering Accreditation Board (or CEAB). CEAB approaches accreditation holistically. A degree program is assessed as a whole where a set number of accreditation units must be approved, and a list of twelve graduate attributes must be met. Ideally, every course in a program contributes to the total number of required accreditation units and the measurement of a few or several graduate attributes. The accreditation units correspond to types of content and are categorized in a high level manner as following [6]:

- mathematics,
- natural sciences,
- engineering science,
- engineering design,
- complementary studies, and
- other unspecified content.

The graduate attributes are also high level and can be thought of as program-level learning outcomes. The twelve graduate attributes are as follows:

1) Knowledge base for engineering;
2) Problem analysis;
3) Investigation;
4) Design;
5) Use of engineering tools;
6) Individual and team work;
7) Communication skills;
8) Professionalism;

9) Impact of engineering on society and the environment;
10) Ethics and equity;
11) Economics and project management; and
12) Life-long learning,

where every one of them can be measured as introduced (I), developed (D), or applied (A) [6].

The school of engineering currently requires the instructor of every course to map the course outline learning outcomes to the graduate attributes. This can be a one-to-one or many-to-one, but not many-to-many relationship. Also, every learning outcome is ideally evaluated in two or more assessment items (e.g., quiz or exam questions, laboratory assignments) or other learning and teaching activities. Basically, evaluating students' performance for a specific learning outcome is used for measuring their achievement level for the corresponding graduate attribute. If, for a particular graduate attribute, a certain percentage (25-30%) of the students in a course exhibit unsatisfactory performance, i.e., they do not meet a minimum required standard, extra measures must be taken. For example, the instructor may be asked to devise a plan of improving the course for its next offering.

Table 2. ABET student outcomes [7] and their equivalent CEAB graduate attributes

No.	Student outcome	Equivalent graduate attribute (s)
1.	an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics	2); 1) is implied
2.	an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors	4)
3.	an ability to communicate effectively with a range of audiences	7)
4.	an ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts	8), 9), and 10)
5.	an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives	6) and 11)
6.	an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions	3); 5) is implied
7.	an ability to acquire and apply new knowledge as needed, using appropriate learning strategies	12)

Note that in the United States the organization equivalent to CEAB is the Accreditation Board for Engineering and Technology Inc. (ABET). Many aspects of the accreditation performed by these two bodies are quite similar (e.g., length of an accreditation cycle, campus visits, program evaluators, expectations for a certain number of credits in the natural or physical sciences, mathematics, and a final-year capstone design project, etc.). Two distinctions, however, can be made between the two accreditation bodies. One is that graduate attributes in the United States are referred to as ‘student outcomes’. The seven student outcomes formulated by ABET [7] and their equivalent CEAB graduate attributes are listed in Table 2. These seven student outcomes may be complemented by additional outcomes articulated by a particular program [7]. The other distinction is a criterion by ABET referred to as ‘program educational objectives’, which is again something published locally by a particular program. As far as the authors are aware this criterion does not exist in Canada.

Methodology

This section describes the course design and development, some implementation aspects, and some of the observations collected by a research assistant. The next three sub-sections address these matters.

Course design

The numerical methods course was designed with alignment in mind. More specifically, the alignment was between the graduate attributes, the course outline learning outcomes, any learning and teaching activities, and the assessment components. For example, the course had four high-level learning outcomes listed in the course outline (see Table 3). Each one of those learning outcomes was created to taget a specific graduate attribute and with the intention of covering as much of the congnitive spectrum in Bloom’s taxonomy as possible (again, see Table 3). The learning and teaching activities in class and the homework review question sets were designed to address the learning outcomes and to also prepare the students for the graded assessment items. The graded assessment items focused on the learning outcomes and included quizzes, laboratory assignments, a final project, and a final exam. Altogether there were five cycles of: a review question set, followed by an individual and group quizzes, followed by a laboratory assignment submission. More details on the in-class activities and the assessment items can be found in the next sub-section.

Table 3. Course outline learning outcomes for the numerical methods course and their associated graduate attributes and corresponding Bloom’s taxonomy cognitive levels

#	Course outline learning outcome	Graduate attribute	Level in Bloom’s taxonomy
1.	Apply general numerical methods to solving complex engineering modelling and computational problems	1 (D)	1-3
2.	Assess the performance of numerical methods in terms of errors and applicability	2 (I)	3-4
3.	Point out advantages and disadvantages of different numerical algorithms for a given engineering problem	3 (D)	4-5
4.	Program numerical methods proficiently in MATLAB in order to solve engineering problems	5 (A)	3, 6

The course included a significant amount of group work. For example, the group quizzes, the laboratory assignments, and the final projects were all done in groups. Working in groups aimed at tackling the issue that some students had previous programming experience in MATLAB, while others did not. In fact, when generating the groups, the instructor intended to group students with a diverse skillset, so that they can complement and learn from each other. The skills the students were surveyed on were: 1) experience with MATLAB; 2) experience with other programming languages or in general interest in algorithms and mathematics; 3) technical writing skills; and 4) oral presentation skills. The onus was on the students to distribute the work among their group mates. In order to mitigate any potential conflicts the groups were asked to draw up a group contract and also to distribute their grade according to the time and effort put in by each group member.

Course implementation

This sub-section includes more detail on how the lectures were run. It also explains how the review question sets, the quizzes, the laboratory assignments, and the final project were implemented.

Lectures: During the lecture periods most of the material was presented in an inductive manner [8]. First, a question or a problem was given to the students, and then the theory behind how to solve the problem was covered. The instructor attempted to integrate theory and practice so as to keep the students interested in the subject matter and motivated to study. Also, the instructor frequently performed MATLAB demos for the students.

Review question sets: After a course module or modules were completed, a set of review questions was released for the students. The review question sets included questions from the textbook along with their final answers, but with no solutions. The review questions were meant to be solved on paper with pen/pencil and a basic scientific calculator in preparation for the quizzes and the final exam. Students were expected to attempt the questions on their own and submit their work for completion marks.

Quizzes: The submission of a review question set was followed by individual and group quizzes inspired by team-based learning (TBL). The groups consisted of three or four students. The quizzes included essential conceptual questions in the true/false or multiple choice formats followed by written and/or calculation questions. Also, note that they were written during class time. The quizzes were run in the following manner: first, students were handed out the individual quizzes and were given 20-25 minutes to complete them on their own; then, the individual quizzes were collected by the instructor, each group was handed out the same quiz and was given another 20-25 minutes to complete it together. In theory, the quizzes are supposed to be low impact assessments allowing students to learn during the group discussion even or especially if they have made mistakes during the individual phase. Such TBL exercises are meant to increase student retention [9] and engagement [10], [11]. The following formula is used in this paper for estimating the short-term gain factor for a particular group:

$$Gain = {}^{G\ [\%]}\!\big/_{I\ [\%]}$$

where ”G” is the grade for the group quiz and ”I” is a grade from the individual quiz. If the minimum individual grade is used for a particular group, the formula would yield the maximum gain factor (i.e., the gain factor for the weakest student in a particular group). If the maximum individual grade is used, then the result would be the minimum gain factor (i.e., the gain factor for the strongest student in a particular group). Finally, if the median individual grade is used – the median gain factor would be estimated.

Laboratory assignments: During the lab time, i.e., concurrently to the review question sets and the quizzes, students were given two lab periods to complete an assignment in the same groups. Altogether, there were five laboratory assignments in the course:

1) Introduction to MATLAB and numerical error
2) Solving non-linear equations and systems of linear equations
3) Interpolation and curve fitting
4) Numerical differentiation and integration
5) Ordinary differential equations (ODEs) – initial value problems (IVPs)

The purpose of the laboratory assignments was to reinforce the concepts learned and to provide students with hands on experience. More specifically, students had to implement algorithms in MATLAB, deal with larger data sets, and experiment with multiple ways or attempts to solving a problem. For example, for each of the problems in the assignment the groups had to come up with a main solution in MATLAB, but also with an alternative solution in order to conduct an independent check of their work. The learning outcomes associated with each laboratory assignment are listed in Table 4. Table 4 also links the assignment learning outcomes to the course outline learning outcomes they were supporting. It can also be seen that the Bloom's taxonomy congnitive levels three to six were targetted. Note that levels one and two were implied.

Table 4. Laboratory assignment learning outcomes for the numerical methods course in support of specific course outline learning outcomes and their corresponding Bloom's taxonomy cognitive levels

Lab#	Laboratory assignment learning outcomes	Course outline learning outcomes	Highest level in Bloom's taxonomy
1.	• Write a MATLAB script to perform repetitive computations • Calculate true and estimated relative errors • Differentiate between different types of errors in numerical solutions such as rounding / chopping and truncation errors • Identify particular cases where numerical error has "catastrophic" effects • Conduct independent checks of numerical solutions	4 1 2 2 3	3 3 4 4 4-5
2.	• Solve non-linear equations via bracketing and open methods • Compare the efficiency of the bisection and regula falsi methods • Compare the convergence of the fixed-point iteration and Newton-Raphson methods	1 2 2	3 4 4

	• Solve systems of linear equations via direct and iterative methods • Implement numerical methods for solving non-linear equations and systems of linear equations by writing functions / scripts in MATLAB • Conduct independent checks of numerical solutions	1 4 3	3 6 4-5
3.	• Search for a certain number of data points closest to a data point of interest • Apply Lagrange interpolating polynomials based on a certain number of “errorless” data points • Construct splines and use them to perform interpolation • Fit redundant data to a mathematical model using linear least squares regression • Implement numerical methods for interpolation and curve fitting by writing functions / scripts in MATLAB • Conduct independent checks of numerical solutions	4 1 1 1 4 3	3 3 3 3 6 4-5
4.	• Apply finite differencing methods on tabulated data for the purpose of numerical differentiation • Apply finite differencing methods for the numerical differentiation of a function • Use Richardson’s extrapolation and Romberg integration for the purposes of numerical integration • Implement numerical methods for numerical differentiation and integration by writing functions / scripts in MATLAB • Conduct independent checks of numerical solutions	1 1 1 4 3	3 3 3 6 4-5
5.	• Solve a single ODE via various numerical methods such as Euler’s, the modified Euler’s, midpoint, and the classical 4th order Runge-Kutta methods • Decompose a second-order ODE into a system of two first-order ODEs • Experiment with an iterative predictor-corrector method • Assess the usefulness of implicit methods in terms of the stability of the numerical solution • Implement numerical methods for the solution of a first order ODE or a system of first order ODEs by writing functions / scripts in MATLAB • Conduct independent checks of numerical solutions	1 1 3 2 4 3	3 3 4-5 4 6 4-5

Final project: The purpose behind the final project was to bring more relevance to the material taught in the course. As mentioned in the introduction, the students enrolled in the course come from three different departments. Even though methods learned in the course are and/or can be used in a lot of follow-up courses, on paper this is a terminal course. As such, it appears as though some students put in less work in it compared to other courses which are listed as prerequisites. In order to break this stigma, the instructor ran a final group project, where the project topics were student-led. The students were supposed to contact professors in the faculty of engineering and inquire about projects in research and industry where numerical methods are applied. In terms of components, the final project was broken down into a short proposal, an oral presentation during the last week of classes, and a final technical report due before the final exam.

Observations by a research assistant

In order for the instructor to receive third-party feedback on the learning and teaching effectiveness in the course, a research assistant attended a limited number of contact hours. The research assistant collected three types of data: 1) after two of the lessons she asked the students to write a minute paper on the three muddiest concepts in that lesson; 2) after two of the course modules she asked the students to fill out an end-of-unit survey where they had to pick the three hardest concepts and the three concepts they had managed to master from a concept inventory for that module; 3) she performed in-class observations during three periods: a standard lecture followed up by a MATLAB demo, a laboratory session, and a group quiz. For the purpose of the in-class observations an observation protocol was filled out each time. The protocol intended to capture various aspects of active learning including constructive and interactive learning [12], [13]. Essentially, the research assistant had to track the teacher's actions and the students' engagement in two-minute increments. The most important feedback the instructor was looking for was on the MATLAB demos and the group quizzes.

Discussion

Below is a set of reflections by the instructor. Following that is the feedback from the third-party research assistant.

Instructor reflections

Group work: Introducing group work into this course definitely made improvements. The main reason for this statement is the fact that before instructors received many complaints about the MATLAB programming aspects of the course (e.g., civil students not able to cope, while mechanical students not given the chance to improve), but no such complaints or tension was experienced in the spring intersession of 2019. Students who did not have any MATLAB background were either able to pick up the language without being stressed that they were on their own or they managed to contribute to the assignments by conducting the alternative solutions. A tricky thing about the group formation was that a number of students dropped the course after classes had already started and after they had already been assigned a group. Thus, some groups were left with three members while the majority of the groups had four. Also, while the idea was to generate groups with a diverse skill set, some students simply chose to work with their friends. The main author does not foresee an issue with incorporating group work in a much larger class other than increasing the administrative overhead for which a course coordinator would be responsible. In future course offerings a learning outcome related to graduate attribute #6 (Individual and team work) should be added to the course outline.

Review question sets: Originally, the review question sets were not supposed to be graded items. Students in a previous course, however, had expressed that even if they were worth a nominal amount that would serve as a motivating factor for them to complete them. At the end, they were worth 1% each and were graded for completion only. The sad part about this was that a small minority of the students would occasionally submit simulated work, i.e., either an empty .pdf file or a "fake" solution with boxed final answers at the bottom of each page. Grading the review question sets in a larger class would be prohibitively expensive in terms of time spent by the instructors and the TAs. So there are ultimately two options for implementing this with a much larger class size: 1) the questions are provided for practice only, i.e., they are not graded at all; or 2) the questions are run through an online system which has the capabilities of numerically randomizing the questions and grading them automatically.

Quizzes: The group quizzes seemed to be a success as they generated a lot of lively discussions. Table 5, Table 6, and Table 7 show the maximum, median, and minimum gain factors for the five quizzes conducted in the course. The maximum gain factor is arguably too optimistic. The median and minimum gain factors are probably more realistic estimates of the improvement in student performance. Note that, even though it is generally expected that the gain factor would be equal or greater than one, sometimes this is not the case. There appear to be situations where the highest individual grade is higher than the group grade. The speculation here is that the student with the highest grade got a higher grade compared to the group grade either by chance or they were not able to defend their answers to their team mates during the group discussion. A big advantage of running the quizzes during class time was an improvement in class attendance at least on the days with quizzes. A downside was that even though there was a significant amount of class time left after the quizzes were over some students would just pack up and leave the classroom while the instructor was teaching. The instructor found this was disrespectful to both him and the rest of the students. Also, the individual quizzes were tough to invigilate. They were only worth 2.5% each, so one would not expect students to risk cheating for such low impact course components. There were, however, a number of cases where students were exhibiting questionable behaviour during the individual quizzes. The instructor had to resort to asking students to change their seats. Scaling this from one section with 65 students to two sections with 150 students each would also require a significant instructor time commitment as there will have to be two versions of five quizzes run on different days. In addition, there may be issues with students requiring to write exams with accommodations.

Table 5. Results for the maximum gain factor

	Quiz 1	Quiz 2	Quiz 3	Quiz 4	Quiz 5
Min	1.10	1.25	0.96	1.04	1.00
Average	1.23	1.88	1.44	1.87	1.65
Max	1.41	2.36	2.17	4.80	5.60

Table 6. Results for the median gain factor

	Quiz 1	Quiz 2	Quiz 3	Quiz 4	Quiz 5
Min	1.00	1.05	0.90	0.96	1.00
Average	1.08	1.43	1.18	1.16	1.08
Max	1.24	1.75	1.63	1.57	1.30

Table 7. Results for the minimum gain factor

	Quiz 1	Quiz 2	Quiz 3	Quiz 4	Quiz 5
Min	0.90	0.91	0.81	0.92	0.92
Average	1.02	1.18	1.02	1.00	1.00
Max	1.14	1.47	1.24	1.09	1.13

Laboratory assignments: The laboratory assignments seemed to be helpful. The students who attended the lab periods asked a fair amount of questions. They also seemed to improve their technical writing skills. After the first lab, the instructor had to include notes on what should be the structure of a technical report and how to do basic formatting. Apparently, many students use web-based word processors with limited capabilities in high school. So when they come to university they do not know how to properly typeset a technical report. For example, they have trouble with styles, with automatic numbering and table of contents generation, and with figure and table captioning. The main author does not foresee any issues with the laboratory assignments in a larger class setting. Also, in future course offerings a learning outcome related to graduate attribute #7 (Communication skills) should be added to the course outline.

Final project: Surprisingly, students did not complain about the course load (especially the one due to the final project) in the course evaluations. However, while a good idea, the final project was very time consuming to assess. First the instructor had to screen the proposals and provide the groups feedback on their project topics. Then, the instructor and the TAs had to sit through all presentations and evaluate them. Finally, the instructor had to grade the final project technical reports. Due to the wide range of topics, the reports and presentations were graded based on a holistic rubric. This is probably the hardest aspect of the course when it comes to scaling it up to a much larger class size. Now instead of dealing with 13-20 project proposals, oral presentations, and projects, the instructors and TAs will have to handle 25 to 50 of them per section. Even if the oral presentation component was dropped (which is not advisable) some of the grading will have to be transferred to the already busy teaching assistants. Some major structural changes to the course will have to be made if a final project is incorporated in the regular course offerings. For example, the final project can replace the final exam.

Research assistant feedback

The research assistant provided feedback on the three observed contact periods, i.e., a standard lecture followed up by a MATLAB demo, a laboratory period, and a group quiz. Her reflections, geared specifically towards the learning and teaching effectiveness of the course, are listed next.

MATLAB demo: According to the research assistant there was a steep decline in student engagement right after the instructor finished lecturing and started the MATLAB demo. Only about 10% percent of the students had their own laptops and were following the demo. Many of the students zoned out relying that the source code will be posted on the learning management system. In fact a few students at the back of the classroom were falling asleep after the lights were turned off to increase the projector brightness. Overall, the research assistant found that the demo was least effective compared to the other two observed contact hours.

Laboratory period: The research assistant stated that the laboratory period went in a fairly standard manner for a computer lab. An interesting note she made was that the TAs did not interacting with the students as much as the instructor.

Group quiz: According to the research assistant the group quiz prompted the most opportunity for constructive / interactive learning. Students were discussing and trading ideas while solving a problem together, i.e., in a similar manner as to how things happen in the real world. She also thought that the group quiz eased the anxiety as she could no longer sense any feelings of nervousness in the students. There were some students who did not engage (e.g., one in four in the occasional group), but all in all this was the most effective learning period that the research assistant observed.

Summary and conclusions

This paper described the design and implementations in an engineering numerical methods course. Changes made in the course were aimed at improving student experience while at the same time fulfilling accreditation requirements and aligning learning outcomes with class-time activities and graded assessments. The students were divided into groups where the members of each group were meant to have a well-rounded set of skills as a whole, e.g., programming experience in MATLAB, interest in math, technical writing skills, and oral presentation skills. The submissions throughout the semester were clearly separated into a set of conceptual self-assessment / review questions (i.e., homework assignments done individually), and programming exercises (i.e., labs primarily done in the tutorial contact hours and requiring group reports). The course also included a student-lead/inquiry-based final project (broken down into a project proposal, a group oral presentation, and a technical report) in order to inspire the applicability of the learned material. Finally, the lecture contact hours were meant to include multiple active learning strategies, such as team-based learning inspired quizzes, solving example problems blended with theory, and performing MATLAB demos. The paper also included a third-party evaluation of the learning and teaching effectiveness of some of the developments, where a research assistant conducted several in-class observations, lesson minute papers, and end-of-unit surveys.

Overall, the group work went well especially the labs and the group quizzes. The labs in particular could also be easily scaled up in another offering of the same course with a larger class size. The final projects and the review questions sets, however, were time consuming to grade. The review question sets were only graded for completion, and the project oral presentations and final reports were graded with a high level rubric. Those two aspects will also be the most expensive in terms of time and effort if scaled up for a much larger class size. The review questions will require a system which can grade automatically, and the final projects may not be feasible unless either more TAs are involved in the grading or the project entirely replaces the final exam. The instructor will continue trying to incorporate more active learning in his lectures. However, the MATLAB demos should be limited to only running short live scripts. Also, with having many review question set – quiz – lab cycles, group work, deadlines involving the final projects, etc., the instructor is recommending having a lecture plan for the semester where any modifications to the course schedule are frequently updated and shared with the students. Also, it is recommended to have an extra document in addition to the course outline where housekeeping and administrative aspects of the course are explicitly put down in writing (e.g., the expectations for the groups and individuals within groups). The next big challenge for this course will be to attempt running as many of the changes described in this paper as possible in the regular large-class offering.

In terms of future work, the authors plan on searching the literature for methods or tools on group formation and other aspects of team work. An example of such a tool is the Comprehensive Assessment of Team Member Effectiveness (CATME) [14]. Also, the authors would like to investigate other existing observation protocols such as COPUS, which stands for Classroom Observation Protocol for Undergraduate STEM (i.e., post-secondary science, technology, engineering, and mathematics courses) [15].

References

[1] B. S. Bloom, "Taxonomy of educational objectives," 1956, Accessed: Oct. 01, 2019. [Online]. Available: http://agris.fao.org/agris-search/search.do?recordID=US201300633852.

[2] D. R. Krathwohl, "A Revision of Bloom's Taxonomy: An Overview," *Theory Into Practice*, vol. 41, no. 4, pp. 212–218, Nov. 2002, doi: 10.1207/s15430421tip4104_2.

[3] M. Prince, "Does Active Learning Work? A Review of the Research," *Journal of Engineering Education*, vol. 93, no. 3, pp. 223–231, 2004, doi: 10.1002/j.2168-9830.2004.tb00809.x.

[4] S. Freeman *et al.*, "Active learning increases student performance in science, engineering, and mathematics," *Proceedings of the National Academy of Sciences*, vol. 111, no. 23, pp. 8410–8415, Jun. 2014, doi: 10.1073/pnas.1319030111.

[5] L. Michaelsen, A. Knight, and L. Fink, *Team-Based Learning: A Transformative use of Small Groups in College Teaching*. 2004.

[6] Engineers Canada, "Consultation Group - Engineering Instruction and Accreditation." Jan. 07, 2016, [Online]. Available: https://engineerscanada.ca/sites/default/files/Graduate-Attributes.pdf.

[7] ABET, "Criteria for accrediting engineering programs." Nov. 30, 2019.

[8] M. J. Prince and R. M. Felder, "Inductive Teaching and Learning Methods: Definitions, Comparisons, and Research Bases," *Journal of Engineering Education*, vol. 95, no. 2, pp. 123–138, 2006, doi: 10.1002/j.2168-9830.2006.tb00884.x.

[9] M. J. McInerney and L. D. Fink, "Team-Based Learning Enhances Long-Term Retention and Critical Thinking in an Undergraduate Microbial Physiology Course," *Microbiol Educ*, vol. 4, pp. 3–12, May 2003.

[10] M. C. Clark, H. T. Nguyen, C. Bray, and R. E. Levine, "Team-Based Learning in an Undergraduate Nursing Course," *J Nurs Educ*, vol. 47, no. 3, pp. 111–117, Mar. 2008, doi: 10.3928/01484834-20080301-02.

[11] B. M. Thompson *et al.*, "Team-based learning at ten medical schools: two years later," *Medical Education*, vol. 41, no. 3, pp. 250–257, 2007, doi: 10.1111/j.1365-2929.2006.02684.x.

[12] M. T. H. Chi and R. Wylie, "The ICAP Framework: Linking Cognitive Engagement to Active Learning Outcomes," *Educational Psychologist*, vol. 49, no. 4, pp. 219–243, Oct. 2014, doi: 10.1080/00461520.2014.965823.

[13] M. T. H. Chi, “Active-Constructive-Interactive: A Conceptual Framework for Differentiating Learning Activities,” *Topics in Cognitive Science*, vol. 1, no. 1, pp. 73–105, 2009, doi: 10.1111/j.1756-8765.2008.01005.x.
[14] M. L. Loughry, M. W. Ohland, and D. J. Woehr, “Assessing Teamwork Skills for Assurance of Learning Using CATME Team Tools,” *Journal of Marketing Education*, vol. 36, no. 1, pp. 5–19, Apr. 2014, doi: 10.1177/0273475313499023.
[15] M. K. Smith, F. H. M. Jones, S. L. Gilbert, and C. E. Wieman, “The Classroom Observation Protocol for Undergraduate STEM (COPUS): A New Instrument to Characterize University STEM Classroom Practices,” *CBE Life Sci Educ*, vol. 12, no. 4, pp. 618–627, 2013, doi: 10.1187/cbe.13-08-0154.