6

Student Activation in a Large Bachelor-level Math Course

Mohammad Sadegh Talebi

Department of Computer Science
University of Copenhagen

Introduction

Bachelor-level introductory science courses have been mostly taught using traditional lectures, often referred to as 'live' lectures, an approach which existed since early days of modern academic systems and continues to predominate other alternatives. Its most classical form was delivered using blackboard presentations, which were superseded by various forms of slide presentation, e.g., overhead projectors and modern computers. Modern technologies continue to make computer-assisted slide presentations a much more attractive way to students and instructors alike, in particular, in large courses with 100+ students. Although modern slide presentations gave a remarkable boost to traditional blackboard lectures, the efficacy of such lectures, irrespective of the media used, has been questioned by many researchers; see, e.g., Mazur, 1997. Traditional lectures are argued to suffer from several drawbacks, which could negatively impact the learning process. A typical and recurrent problem is low participation. Another notable drawback is that students could lose the thread at some early point[1], thereafter they follow less and less (Mazur, 1997). Yet another one is that maintaining a good and universally accepted lecturing pace is difficult, if not impossible: some students find it slow and boring, and others simply cannot follow the class.

A well-recognized approach departing from the traditional lecturing system is *flipped classroom*, also known as "inverted classroom" or "re-

[1] This is context-dependent, but a rule of thumb says it happens after 20 minutes.

versed instruction", which aims to shift the teacher-centered learning process to a student-centered one (Bergmann & Sams, 2012; Tucker, 2012). There seems to be little consensus on the definition of the flipped classroom (Schell & Mazur, 2015), but its most universal definition reads (Bergmann & Sams, 2012): "Basically the concept of a flipped class is this: that which is traditionally done in class is now done at home, and that which is traditionally done as homework is now completed in class". Some researchers argue that flipped classroom is a more a mindset than an approach; see, e.g., Schell and Mazur, 2015. The flipped classroom approach admits a broad range of implementations, but most of which strive to use tools to activate students in the class, and thus promote students' involvement in the learning process. In doing so, *peer instruction* is a prominent pedagogical technique (Lymna, 1981; Mazur, 1997), which has been around for more than three decades. Peer instruction allows for more interaction between both peers (i.e., students) and student-teacher, and whose key target is to activate all participants in the class making them involved in the learning process. There is a rich literature reporting the success of peer instruction in enhancing the learning performance in various courses; see, e.g., Mazur, 1997, Cortright et al., 2005, and Giuliodori et al., 2006.

This study presents an alternative design for a large bachelor-level course with the aim of activating students, and ultimately promoting their involvement in the entire learning process. The course under study is 'Introduction to Numerical Methods' (abbreviated hereafter as *NumIntro*), which introduces students to the realm of numerical analysis, and thus, plays a vital role in applications of mathematics in practice. It is essentially relevant in every situation where some form of computation is performed in a computing system, or where some mathematical model is implemented in real-life. It is therefore an important course bridging theory and practice, and has thus both practical and theoretical elements.

My aim in this project is to present ideas on how to restructure the basic course elements of NumIntro so that the teaching approach becomes more aligned with the flipped classroom approach. In so doing I make use of several core in-class activities carefully designed to activate students in the class. The common aspect of the core activities is their aim to boost students' involvement in the class, while some of them implement some degree of peer instruction. The core activities incorporate feedback from departmental and educational supervisors when trying similar activities in another course of similar nature, but also consider students' evaluations of the last edition of the course. At last, the choice of activities and students' involvement are supported and discussed considering relevant literature.

Context of the Study: Course Description of NumIntro

Style and Organization

Offered by the Department of Mathematical Sciences (MATH), NumIntro is a 7.5 ECTS bachelor course taken by students from various disciplines including mathematics and physics (course website: https://kurser.ku.dk/course/NMAA09005U). It is a large bachelor-level math course, typically with 100+ students. In the 2020 edition of the course, 135 students signed up for the course, most of whom successfully completed the course. The course is usually taught by two instructors. Its 2020 edition was taught by Mogens Bladt (Professor, MATH) and myself (Assistant Professor, DIKU). It also had four teaching assistants (TAs). The course materials and activities can be categorized into two parts: theory and programming. The programming language taught in the course is Python, a high-level computer programming language widely used in scientific computing. Offered to a wide range of students, NumIntro assumes no prior background on computer programming.

The course takes place in Block 1, and I am responsible for the first 3 weeks of the course, a total of 6 sessions. Conventionally, each of these sessions is divided into two parts: the first part is devoted to theoretical aspects of numerical computation whereas the second deals with teaching the basics of computer programming (Python) to implement numerical methods in computer. The rest of sessions, taught by the other instructor, mostly deals with the theory part. Following each lecture, there are a few (in 2020, four) parallel, identical exercise sessions where TAs solve pre-assigned theoretical and programming exercises. Each TA is responsible for a pre-assigned group of around 35 students. TAs also help students debug their Python programs.

NumIntro's Intended Learning Outcomes (ILOs)

According to NumIntro's course description, a student, upon completion of the course, acquires knowledge on (a) standard numerical methods for solving equations, approximation of functions, integration and differentiation, etc., and (b) simple programming in an imperative language (in recent years, Python), including procedures/functions, variable, sentences, numerical expressions, scope (and beyond). More precisely, NumIntro strives at achieving the following:

"On the completion of the course, the student, will be able to: (i) explain what distinguishes "exact mathematics" from "numerical mathematics", (ii) set up standard methods for numerical solution of non-linear equations, linear equation systems and eigenvalue problems, (iii) set up simple models for approximation of functions, differential quotients and integrals, (v) use an imperative programming language to write and execute small programs, and in particular, implement and solve methods in (i)-(iii) using that language."

The theory part of NumIntro deals with methods geared with the practical aspects. This feature together with the programming part make NumIntro a bit special, and perhaps not directly comparable to most other bachelor-level math/physics courses.

Assessment

The assessment consists of a three-hour written exam and a compulsory hand-in, each counting for 50% of the total grade. The compulsory hand-in is a one-week individual take-home exam, usually taking place in Week 6, and comprises a well-designed theoretical problem followed by some relevant programming tasks. In contrast, the written exam only contains theoretical questions. Prior to the written exam, the students receive written feedback on their compulsory hand-in.

The Last Edition of NumIntro

In pre-2020 editions of NumIntro, the theory lectures followed a traditional classroom approach, mostly using a blackboard. This was no longer the case in the last edition of NumIntro (in 2020), which was taught entirely online due to restrictions posed by the Covid-19 outbreak. In this last edition, which was my first experience with NumIntro, I organized each session into two parts: First, I would give a live slide presentation over Zoom (Zoom Video Communications, 2021) to teach the theory part. It was followed by a live programming session, where I would first introduce some concept of Python orally (i.e., with no slides) and then directly examine them in computer. Live programming would be implemented in a proper pace so that student could do programming simultaneously. For example, I would teach how to define a variable and do some small calculations. Then I would write a small script and run it, and so would do the students. Recorded videos of both parts made available after each session. The other instructor took a different approach however: He divided each session into several sub-sessions, where for each he recorded a short video clip (10-20 minutes long). In the videos, he used slide presentation trying to mimic the step-by-step derivations of mathematical formulas as in blackboard presentation. These pre-recorded videos were made available on Absalon one week before the class. He would then organize a one-hour Q&A session (in Zoom) in each week, where students could show up and ask questions about the covered topics.

The course implementation also featured an element which was not affected by the outbreak: Prior to each week, we posted a weekly slip ('ugeseddel') listing: (i) the topics to be covered in both theory and programming parts to a detailed extent as well as their corresponding reading assignments from the textbook; (ii) some theoretical exercises, mostly selected from the textbook; and (iii) programming exercises aiming to implement theoretical part taught within each week. The students were encouraged, but not obliged, to read the materials before the class and try to solve the exercises.

Lessons Learned

Each instructor had the complete freedom to choose his preferred mode of teaching, and trying different approaches were by no means coordinated to, e.g., boost the learning performance or conduct some experiments to identify the superior one. Nonetheless, students' evaluations revealed that they preferred pre-recorded videos to live lectures. Under normal circumstances where student life is not affected by the quarantine, this observation is consistent with what the literature reports. This was nonetheless surprising to me as I expected that a live lecture could to a great extent resemble a traditional lecture in a physical classroom, of which students were deprived. I could argue that the lonesome student life arising from the Covid-19 quarantine could be remedied when the students virtually gather, and this helps amend sense of community, which has become fragile in the quarantine days. Students' feedback proved otherwise, however.

Description of the Activity

The 2020 edition of NumIntro turned out to be a valuable experiment revealing some precious and encouraging results. These key observations motivated us to adopt a flipped classroom approach for the entire course in the next edition (in Block 1, 2021). Although one instructor followed a flipped classroom approach in the last edition, the involved Teaching Learning Activities (TLAs) were still mostly *teacher-managed,* in view of the terminology in Biggs and Tang, 2011. In order to enhance the learning performance, I propose to use some well-tailored and solid activities. This section aims to make these activities more precise. As prepatory task, for each session, there will be some reading assignment and the main material will be catered in the form of a series of 10-20-minute-long video lectures, which be uploaded to the course page in Absalon a few days before the class. Reading assignments are considered optional, however. In contrast, watching videos is compulsory as they constitute the main pre-class teaching element. The topics in both theory and programming parts lend themselves very well to this implementation. In particular, for the programming part, pre-recorded videos cover definitions, language constructs and commands, and other syntactical elements, as well as some small programming examples. At the heart of my proposal are a number of core activities designed to be conducted in the class, which I shall generically call *in-class activities*. I specifically consider 4 in-class activities discussed below.

In-class Activity 1: QnA

The first core activity is Question and Answer (QnA), in which students are given the opportunity to ask questions related to the topics covered in the reading assignment and pre-recorded videos. The length of QnA could be adaptively chosen depending on, among other things, the number of questions as well as the difficulty and importance of the topic covered. Although this is an in-class activity, the students are also encouraged to pose questions in advance of the class via the features available through the course page. QnA might sound dull, and its implementation may seem trivial. However, it is crucial to be included in each session, and its implementation involves some subtly. Dobbing QnA a separate activity helps make its importance more salient to the students.

In-class Activity 2: Conceptual Quiz

The second activity is what I call conceptual quiz – though the terms may already exist in the literature. It features a quiz comprising a few multiple-choice questions or questions with short answers, which is supposed be completed by students individually. It is not supposed to deal with complicated math/programming exercises, but rather with questions related to core concepts covered. There are several ways to implement this, and nowadays this activity can be administered easily and cheaply, thanks to the widespread use of online platforms tailored to this purpose, such as Padlet (Wallwisher Inc., 2021) and Socrative (Showbie Inc., 2021). These platforms also allow for preserving anonymity and ease in contrasting responses. I intend to use such platforms to implement quizzes in a sequential fashion, where questions will appear one by one on the screen and for each, students, who are connected to the relevant platform, will be given some short time to commit to an answer (in the case of multiple-choice question) or write a short answer otherwise. Then, the correct answer together with a short supporting discussion will appear on the screen (in a physical or virtual class), while the instructor will elaborate on the correct answer and perhaps contrast it with some potential wrong answers. I stress that my 'Conceptual Quiz' is not identical to 'ConcepQuiz' of Mazur (Mazur, 1997), despite the similarity between the two terms.

In-class Activity 3: Exercise Solving in Breakout Sessions

As its name suggests, this activity involves solving a theoretical exercise in a breakout session. In a physical class, a breakout session involves discussion among neighbors, whereas in an online class, this involves discussion among students put in some virtual room. For instance, in Zoom, this can be implemented using the breakout room function. Students will work with their neighbors on a well-designed exercise. First, a theoretical exercise related to the covered topic is presented (e.g., via slide presentation or blackboard). Once the task is clear, the instructor lets the students to reflect and work on it together with their neighbors, collaboratively. Each group is supposed to report its solution to some shared platform (e.g., via Googledoc). Once breakout sessions are closed, the instructor discusses the solution and provides feedback on the reported solution by each group. The sessions could then resume for further discussion among peers. This structure is consistent with the various phases of the TDS model, that is, Devolution, Activation, Formulation, Validation, Institutionalisation (Brousseau & Balacheff, 1998).

In-class Activity 4: Live Programming

This activity consists of on-the-fly programming in the class in order to *individually* solve a programming exercise. More specifically, first the instructor presents a programming task related to the covered topic. The students will be given some time to reflect *in small groups*, but then are asked to individually write a computer program solving the task. Once the time is over, the instructor presents a possible solution (a computer program) in the class, visible on the screen, and runs the code. Next, the instructor discusses possible errors students may have.

Session Synthesis

Having introduced the core in-class activities, I proceed to present some possible plans for the class (relevant for the first 3 weeks). Each session is divided into three sub-sessions. The first is devoted to QnA, which could last for some time depending on the topics covered. My intent is that QnA precedes the other activities. The second sub-session, which concerns the theory part, is devoted to exercise solving in breakout sessions. Furthermore, in every other lecture, it features a conceptual quiz, too. The third sub-session concerns the programming part and features a series of live programming activities. (For a summary, see Figures 1 and 2.)

Mogens Bladt, the current course responsible for NumIntro, is onboard with the suggested plan and is as eager as me to implement it, thus ensuring the congruence. To respect the didactical contract, at the beginning of the next edition, the students will be informed of restructuring of the course elements.

Discussion and Outlook

Students' evaluation in the last edition revealed their admittance of the flipped classroom approach. For instance, students admired various flexibility degrees freely endowed with pre-recorded video lectures: that they could pause videos to reflect or take some break, to control the pace, and to rewind, if necessary, though they were deprived of asking questions on the fly. Considering this, a flipped classroom approach for the entire course is no doubt worth trying.

before class	in-class		
pre-recorded videos	QnA	exercise solving in breakout session	live programming
	sub-session 1	sub-session 2	sub-session 3

Figure 1. Session design 1.

before class	in-class			
pre-recorded videos	QnA	conceptual quiz	exercise solving in breakout session	live programming
	sub-session 1	sub-session 2		sub-session 3

Figure 2. Session design 2.

The efficacy of the proposed plan will be examined in next edition of NumIntro taking place in Block 1 (2021-2022). However, its design and the choice of in-class activities could be supported in light of existing literature as well as some prior experience of implementing similar activities in other courses. Below, I discuss some positive aspects as well as some words of caution in implementing the plan.

QnA

QnA might seem dull or even redundant, but I argue that it is a crucial and viable in-class activity. The literature on flipped classroom strongly supports activities like QnA, often called Q&A, and it turns out that some variant of Q&A is a recurrent in-class element in flipped courses; see, e.g., Zheng et al., 2020 and Lim et al., 2014. The latter work reports a flipped classroom implementation for two math courses, 'Calculus 2' and 'Nonlinear System Theory', where Q&A was the main in-class activity.

In a live lecture, students ask questions of various kinds: some are superficial whereas some others are deeper and well-elaborated. There always are unclear points in the presented materials, e.g., unclear notation, abbreviation, or some confusion with respect to previously taught materials. Most matters like these are immediately resolvable in live lectures. Although questions during a live lecture tend to interrupt the instructor's flow, most instructors would admire them as they are often feedback resourceful: they indicate whether students are following, but they also give clear pointers to misleadingly presented items. Further, many questions asked during live lectures could be a reflection trigger for the others. It is unfortunate that pre-recorded video lectures lack this feature, despite their promising positive aspects. Therefore, by devoting a sub-session to QnA, I strive to compensate for this lacking element. Devoting a reserved slot for QnA conveys an explicit message to my *video-watchers* that they are not left out, and hopefully encourages them to actively participate in the class (and hence in QnA). Afterwards, the hope is to get some feedback from the students as in live lectures, but it notably gives them some time to formulate their questions more solidly. This is also supported by the literature. For instance, Lim et al., 2014, who uses Q&A as the sole in-class activity for two flipped math courses, report some encouraging observation: Respondents in their course evaluation admire that they reviewed their questions before raising them in the class, which led them to prevent asking questions irrelevant to the class.

Perhaps one overlooked benefit of QnA is that it gives the instructor an opportunity to teach students ways to concretely and precisely formulate their questions. For instance, this can be done by paraphrasing the original question. While in a live lecture, questions are asked orally, here it seems a good idea to benefit from online platforms (even in a physical class) allowing students to post their questions in advance of the class, which brings anonymity as an option. As a final note on the importance of QnA, I refer to some advice from my pedagogical supervisor (Grønbæk, 2021): A key goal of "teaching" is to teach the students to ask questions: encouraging their continual curiosity, helping them delve into the topic, and finally helping them elaborate on their questions with the aim of learning to frame their lack of knowledge into a valid, clear question.

Conceptual Quiz

Quizzes constitute a popular form of in-class activities in a plethora of flipped classroom practices and whose use is supported well by the literature. They have been deployed in a variety of courses, e.g., introductory physics (Mazur, 1997), computing courses (Maher et al., 2015), and evolutionary process (Awidi & Paynter, 2019). In these works, quizzes have been used for various purposes: For example, Mazur, 1997 and Maher et al., 2015 make use of quizzes as a proxy to cheaply and rapidly gather feedback on the number of students completed the reading assignment. The purpose of quizzes may not be just to examine the depth of students' understandings; well-elaborated quizzes can be used as a medium, different than the main lectures, to convey knowledge (e.g., Awidi and Paynter, 2019; Maher et al., 2015). Further, Maher et al., 2015 redistributes quizzes to others for peer grading, a feature absent in my proposal.

Ease of implementation, thanks to availability of online platforms, is perhaps a promising feature of this activity. Prior experience of trying a similar activity in another bachelor course indicates that the participation rate is fairly high, even among those who are reluctant to be active in other activities. Anonymous implementation is a key to increase participation. Perhaps a remarkable feature here is that the cheaply gathered feedback has shown to be effective in discovery and clarification of misconceptions (Maher et al., 2015). At times, such misconceptions are experienced to be difficult to identify otherwise. It is worth remarking that such an activity can be used within live lectures to incorporate a change of mode to boost students' participation and regain their attention.

In view of NumIntro's ILOs, trying too many such quizzes could introduce some incongruence or jeopardize the alignment between assessment and the learning activities. The reason is that it is not always easy to design good questions being well-aligned with the ILOs or the assessment. That's the reason why I tend to limit the frequency of them (e.g., in every other lecture or even less). Finally, it is worth noting that some degree of peer instruction (as in, e.g., Maher et al., 2015) could be easily incorporated into in-class quizzes, though my proposal for Activity 2 does not involve any.

Exercise in Breakout Sessions

This activity is no doubt rewarding and its efficacy for introductory Physics courses, among other introductory science courses, is supported well by a plethora of works, the notable reference being Mazur, 1997. Among the proposed activities, this is the one directly implementing a form of peer instruction and provides the best opportunity for the students to really engage in some problem solving: Not only does it involve *individual reflection* about the task, but it also sets the stage for the students to gain skills on how to collaboratively solve a (theoretical) problem. Prior experience of such breakout sessions shows the success in achieving these goals in similar large bachelor courses (Pedersen, 2021).

Despite all such promises, breakout sessions turn out to act like a double-edge sword: They are arguably difficult to control and could easily become rewardless. One big challenge involved is due to those having little, if not none, incentive to actively participate. Surprisingly, this challenge could turn into a rewarding aspect of such activities (e.g., Lucas, 2009; see discussion below). Assignment of students to groups turns out to be a key factor to determine the level of participation, and in turn, success. Therefore, for sessions with random or blind peer assignments, devising many such activities could be deemed a waste of time.

The literature advocating benefits of breakout sessions, similar to Activity 3, combined with various forms of peer instruction abound, e.g., Lucas, 2009, Gok, 2012, Zingaro and Porter, 2014, to name a few. The positive aspects reported include significant impact of students' self-confidence (Gok, 2012; Zingaro & Porter, 2014), creating a lively classroom atmosphere, and success in engagement of passive students in college math courses (Lucas, 2009).

Among the other in-class activities, breakout sessions are the most time and resource consuming. Students might perform slowly or providing feedback to them in the plenum could take longer than predicted. Therefore, maintaining one breakout session activity with enough feedback time could lead to superior results, in terms of achieving ILOs, than greedily provisioning two or more such activities.

I conclude this note by some remarks. Instructors may be eager to try different modes of teaching, inevitably incorporating too many in-class activities. However, it is critically important to consider well-being of students. For instance, some studies have reported that some students find breakout sessions anxiety provoking. This was the case when they are put to spontaneously formed groups, and they found talking to strangers anxiety provoking (Palner, 2021), which could happen in large courses. Implementing some moderate level of activities could be good but trying too many activities might decrease the learning performance. For instance, students' evaluation in a Physics class shows that students admired a moderate use of class activities, and most respondents would prefer to have some degree of blackboard presentation (Haerter, 2020). Finally, introducing various activities gradually over various editions of the course seems very reasonable but is also advisable to ensure the alignment between ILOs, TLAs, and the assessment.

Acknowledgement

The author would like to thank Jan Sølberg, Esben Trabjerg, and Helena Safavi for their careful readings and constructive comments.

References

Awidi, I., & Paynter, M. (2019). The impact of a flipped classroom approach on student learning experience. *Computers & Education*, *128*, 269–283.

Bergmann, J., & Sams, A. (2012). Flip your classroom: Reach every student in every class every day. *International society for technology in education*.

Biggs, J., & Tang, C. (2011). *Teaching for quality learning at university*. McGraw-hill education (UK).

Brousseau, G., & Balacheff, N. (1998). *Théorie des situations didactiques: Didactique des mathématiques 1970-1990*. La pensée sauvage.

Cortright, R., Collins, H., & DiCarlo, S. (2005). Peer instruction enhanced meaningful learning: Ability to solve novel problems. *Advances in physiology education*, *29*(2), 107–111.

Giuliodori, M., Lujan, H., & DiCarlo, S. (2006). Peer instruction enhanced student performance on qualitative problem-solving questions. *Advances in physiology education*, *30*(4), 168–173.

Gok, T. (2012). The effects of peer instruction on students' conceptual learning and motivation. *Asia-Pacific Forum on Science Learning and Teaching*, *13*(1), 1–17.

Grønbæk, N. (2021). Personal Communication.

Haerter, J. (2020). Student activation in theoretical physics lectures. In F. Christiansen & A. Arias (Eds.), *Improving university science teaching and learning pedagogical projects – pedagogical projects 2019*. Department of Science Education, University of Copenhagen.

Lim, C., Kim, S., Lee, J., Kim, H., & Han, H. (2014). Comparative case study on designing and applying flipped classroom at universities. *International association for the development of the information society*.

Lucas, A. (2009). Using peer instruction and i-clickers to enhance student participation in calculus. *Primus*, *19*(3), 219–231.

Lymna, F. (1981). The responsive classroom discussion. In A. Anderson (Ed.), *Mainstreaming digest*. Univ. of Maryland College of Education.

Maher, M., Latulipe, C., Lipford, H., & Rorrer, A. (2015). Flipped classroom strategies for cs education. *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, 218–223.

Mazur, E. (1997). Peer instruction: Getting students to think in class. *AIP Conference Proceedings*, *399*(1), 981–988.

Palner, M. (2021). Problem based learning in the online classroom. In F. Christiansen & A. Arias (Eds.), *Improving university science teaching and learning pedagogical projects – pedagogical projects 2021*. Department of Science Education, University of Copenhagen.

Pedersen, M. (2021). Expanding student perception of linear algebra. In F. Christiansen & A. Arias (Eds.), *Improving university science teaching and learning pedagogical projects – pedagogical projects 2021*. Department of Science Education, University of Copenhagen.

Schell, J., & Mazur, E. (2015). *Flipping the chemistry classroom with peer instruction. chemistry education: Best practices, opportunities and trends*. Willey Online Library.

Showbie Inc. (2021). Socrative. https://socrative.com

Tucker, B. (2012). The flipped classroom. *Education next*, *12*(1), 82–83.

Wallwisher Inc. (2021). Padlet. https://padlet.com/

Zheng, L., Bhagat, K., Zhen, Y., & Zhang, X. (2020). The effectiveness of the flipped classroom on students' learning achievement and learning motivation. *Journal of Educational Technology & Society*, *23*(1), 1–15.

Zingaro, D., & Porter, L. (2014). Peer instruction in computing: The value of instructor intervention. *Computers & Education*, *71*, 87–96.

Zoom Video Communications. (2021). Zoom – video conferencing, web conferencing, webinars, screen sharing. https://zoom.us/