Teaching Statement

John T. Foster

# Overview

I am a somewhat non-traditional teacher in that I rarely give a full 50/75 minute back-to-the-students blackboard-style lecture, preferring interactive project-based learning whenever possible. I have been an early adopter of the *reverse classroom* approach. I have developed three separate PGE courses (PGE 310/311, PGE 323M, PGE383 - High Performance Computing) using this teaching modality in which all of the traditional lecture materials are posted to YouTube and embedded in a course specific websites that I developed. The students are then required to watch the videos[[1]](#footnote-20) before attending class. Accompanying the videos are automatically graded quiz questions to ensure the students complete the videos and continuously provide a feedback assessment of their progress. The in-class part of the course is then offered in a computer lab where daily exercises are carried out in a way that offers students an opportunity to solve problems on their own with my immediate assistance when they need it. Independent projects replace tests in these courses. Thus far I have 100s of lecture videos for these classes. Interestingly, these videos are being watched more often by others, outside of the formal classroom. They currently have over 230K unique views representing more than 800K minutes of viewing on YouTube from interested people in almost every country in the world. Figure [1](#fig:views) shows the historical accumulation of views for the HPC course. I taught this course for 3 semesters at UTSA and am preparing to teach a similar course in the Spring of 2017 to PGE students.

|  |
| --- |
| Historical *daily* views of HPC course YouTube channel |

I believe this serves as a avenue for reputation enhancement, through others familiarizing themselves with my expertise on this topic and seeking out my other contributions in research, etc. Additionally, it creates a lasting catalogue of valuable reference material for my own students and researchers as well as other students at UT.

This flipped-classroom format may not be appropriate for all classes; however, I firmly believe in providing the students a *menu* of options for learning, especially when technology can provide this at little or no time-cost to me. In the classroom, when delivering traditional lectures I use a tablet peripheral interface to my laptop, where I can either develop lecture notes from scratch on blank paper or I can annotate prepared material in real-time. This makes the transition to prepared slides, demos, videos, web-based materials, or live-coding on the laptop during the lecture seamless. While delivering the lecture, I wear a microphone and screen-capture my computer desktop while simultaneously projecting to the screen in the lecture hall. At the end of class, I post the lecture notes developed in class along with a YouTube video of my lecture. *Every lecture is recorded* and available for the students future reference. I spend a little time post-producing the lecture videos to break-up the full lecture into 5–20 minutes topical segments. This post-production of the lectures makes it easier for the students to find the material they are interested in when reviewing for exams or working assignments. Additionally, this allows me to easily assemble an appropriately timed lecture in the advent of an absence in future semesters. Written comments from students over the years have overwhelmingly expressed gratitude for these reference videos and other materials, as not all students learn in the same manner, it provides a way to accommodate several different learning styles. I have used this methodology in several PGE courses at UT including PGE 379 Advanced Geomechanics[[2]](#footnote-23), PGE 334 Reservoir Geomechanics[[3]](#footnote-25), and PGE 323M Reservoir Engineering III[[4]](#footnote-27). Like the HPC course described previously, these videos are being watched by students around the world and have views cumulatively exceeding 100K. Professors Maša Prodanović and Eric van Oort in the PGE Department have adopted and adapted parts of this methodology after discussions with me regarding the tools I use. Other professors have expressed interest in adopting these techniques, and the possibility of using some of these methods to prepare material for a future online Master of Science degree in PGE has been discussed, including at the PGE 2015 faculty retreat where I gave a live demonstration.

It has been my experience that many, if not most, engineering students are aided by visual learning tools and where appropriate I always try to incorporate these into my lectures. Figure [[fig:newton]](#fig:newton) shows an animated computer visualization of tangent lines being drawn and new approximations being computed as a Newton’s method nonlinear solver converges to a root. This is one of many similar such visual aides that I developed (i.e. wrote the *Mathematica* code) for a course on numerical methods[[5]](#footnote-29), these notebooks are currently being translated to an open-source interactive book[[6]](#footnote-31).

I prefer a grading procedure which emphasizes homework assignments and projects that include applications to real engineering problems that the personal computer is needed to solve. As a working engineer, it was a rare case to encounter a problem that could be easily solved analytically; therefore, the computer plays a vital role in the education of my students just as it would assist a working engineer. I would say the primary educational goal for every course I teach is for the students to become better programmer/problem solvers through the use of scientific computing. I force the students to write computer code for nearly every assignment by designing problems that require iteration or full-field and/or multiple parameter space visualizations that are too difficult or intractable to work by hand. I do not insist they use any particular language or engineering software toolbox, but rather emphasize choosing the right tool for the task by discussing the pro/cons regarding speed, algorithm robustness, and ease of programming with them. Begin proficient in several programming languages and engineering sofware packages (virtually all of the most commonly used ones) allows me to do this with ease. I emphasize good coding practices that make their codes easier to debug. I force them to solve verification problems, and to test their code so that they have confidence that the solutions they turn in are correct. I frequently live-code on my laptop in class to explain not only syntax, but the logic of programming which often is the logic of problem solving. I even discuss software *developer operations* such as agile, testing, and version control when appropriate. I have held ad hoc tutorials outside of class on such topics[[7]](#footnote-33). The students I have taught so far seem to respond well to these course designs, despite the common misconception that “engineers don’t like to code”. It’s possible that engineers do not recognize the full worth of programming when taught as a stand-alone subject, but when taught programming as part of the *art of problem solving*, nearly every student I have encountered develops an appreciation and they have expressed this in written comments during course evaluations.

I firmly believe that, just as no exceptions are ever made for engineering students who must use a slide-rule when preparing exam questions (i.e. it’s an expectation that all students have access to a scientific calculator), we have reached the time when the same philosophy should be applied with respect to the laptop computer. When teaching PGE 334 Advanced Geomechanics in the Spring of 2016, for the first time, I allowed students to use their personal or department provided laptops during exams. In this course, there are a class of problems which require using rotation matrices to perform multiple coordinate transformations for the solution. If the students are asked to solve these problems on an exam, an assessment of their performance is influence by how quickly they can type matrices into their calculators. This is not in anyway an assessment of their understanding. I allowed them to program the coordinate transformations and bring their codes with them to class. I even provided verification solutions so they could check that their code was working properly before the exam. They could simply use the code during the problem solution with little worry about making algebraic or typographic errors and allowing them to focus on their physical understanding of the problem. The overall grades on otherwise very similar exams improved dramatically from the previous semester when I taught this course.

Being formally trained as a mechanician, I have a very different background than a traditional petroleum engineer; however, it is my belief that this skill set adds tremendous value in a future where unconventional resources will play a large role in the petroleum industry. These unconventionals require extensive knowledge of geomechanics to efficiently and safely bring into production. Indeed PGE’s curriculum has already began to reflect this by adding both undergraduate and graduate courses in geomechanics over the last several years (i.e. PGE334 and PGE373, both of which I have taught). Additionally, my background in scientific computing and the numerical solution of PDE’s allowed me to seamlessly begin to teach courses in reservoir simulation (PGE323M).

I have always been rated highly in student evaluations. The worst instructor evaluation I have received was in PGE 334 Advanced Geomechanics during the Spring of 2015 (my first time teaching this course and the largest course I had ever taught at the time by a significant amount). The evaluation was a 3.4 which I improved to a 4.3 the next time I taught the course. I attribute the improvement to better preparation (second time teaching the course), a more knowledgable and efficient use of teaching assistants in a large class, as well as implementing new techniques, such as using laptops for exams as previously discussed. This improvement was in spite of requiring the students to program for every assignment, something they supposedly don’t like to do! Generally my instructor ratings are in the 4’s. Table [1](#tab:summary) provides a summary of my course and instructor ratings since becoming an assistant professor in the Fall of 2011.

With regard to assisting (primarily graduate) students in their learning process, I maintain an active weblog[[8]](#footnote-35) of helpful tips and answered questions. Generally these are related to scientific computing topics. I try to maintain a philosophy where anytime I believe a student’s question will come up again in the future, I write a blog detailing the path forward as opposed to providing an individual answer. This has paid off numerous times where I can simply point the students to a blog post where I’ve already answered their specific or a similar question.

I believe I have demonstrated real innovation in the classroom, notably through the use of technology, and I was awarded the *2015 Society of Petroleum Engineers Faculty Innovative Teaching Award* in this regard. Another accomplishment I am very proud of, but is admittedly not completely my own, was for my Teaching Assistant for PGE 323M in Fall 2015, Nkemakonam Egboga, to be awarded the William S. Livingston Outstanding Graduate Student Academic Employee Award by the Graduate Student Assembly. This award, for which I prepared the nomination package, is given to *one graduate teaching assistant in the entire university*. Nkemakonam was the first PGE student and the first Cockrell School of Engineering student to ever be given this award.

Finally, I believe experience with these innovative approaches to teaching will give me an advantage if called on to participate in the PGE Departments forthcoming online MS program or The University of Texas System’s delivery of online or massively online courses, e.g. the edX program, something I would be interested in doing in the future.

Summary of *Course-Instructor* Ratings

|  |  |  |
| --- | --- | --- |
| **Metric** | **Value** |  |
|  | **UT** | Total[[9]](#footnote-37) |
| Total of students taught in organized courses | **198** | 363 |
| Average instructor evaluation for UG courses | **3.9** | 4.1 |
| Average instructor evaluation for Grad courses | **4.5** | 4.5 |
| Average course evaluation for UG courses | **3.8** | 4.0 |
| Average course evaluation for Grad courses | **4.3** | 4.4 |

Course Schedule by Semester with Number of Students Indicated

|  | UTSA[[10]](#footnote-39)[[11]](#footnote-40) |  |  |  |  | UT-Austin[[12]](#footnote-41) |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course** | **F11** | **S12** | **F12** | **F13** | **S14** | **F14** | **S15** | **F15** | **S16** |
| **ME4603** | 21 |  |  |  |  |  |  |  |  |
| **ME6043** |  | 3 |  |  | 6 |  |  |  |  |
| **ME4953/5013** |  |  | 39 | 48 | 48 |  |  |  |  |
| **PGE379** |  |  |  |  |  | 13 |  | 14 |  |
| **PGE334** |  |  |  |  |  |  | 81 |  | 25 |
| **PGE323M** |  |  |  |  |  |  |  | 65 |  |

1. <http://johnfoster.pge.utexas.edu/HPC/video-lectures/> [↑](#footnote-ref-20)
2. <http://johnfoster.pge.utexas.edu/PGE383-AdvGeomechanics/> [↑](#footnote-ref-23)
3. <http://johnfoster.pge.utexas.edu/PGE334-ResGeomechanics/> [↑](#footnote-ref-25)
4. <http://johnfoster.pge.utexas.edu/PGE323M-ResEngineeringIII/> [↑](#footnote-ref-27)
5. <https://github.com/johntfoster/numerical_methods_book/tree/master/mathematica_notebooks> [↑](#footnote-ref-29)
6. <https://github.com/johntfoster/numerical_methods_book/blob/master/TableofContents.ipynb> [↑](#footnote-ref-31)
7. <http://johntfoster.github.io/posts/git-and-github-tutorial.html> [↑](#footnote-ref-33)
8. <http://johntfoster.github.io/> [↑](#footnote-ref-35)
9. ME4953/5013 was a co-listed graduate and undergraduate class. The majority of the students were graduate students and the course was taught as a graduate class, i.e. there were no special assignments or exceptions made for undergraduates in the course. The evaluations were performed together and combined so I used the results for this course only for the Grad course averages. [↑](#footnote-ref-37)
10. My teaching workload was 1 course per semester for my first 2 years at UTSA [↑](#footnote-ref-39)
11. S13 was a teaching buyout [↑](#footnote-ref-40)
12. My teaching workload was 1 course per semester for my first year at UT-Austin [↑](#footnote-ref-41)