Teaching Philosophy

I would like to join an educational institution that encourages, sustains, and nurtures non-traditional engineering education models. The traditional engineering education process is roughly:

1. Students sit through two years of lectures on fundamentals: calculus, physics, chemistry, etc. (also known as the "weeder" courses).
2. After the foundational knowledge has been gained, the students sit through one to two more years of specialized engineering lectures.
3. Finally, as a capstone in the last year the student is tasked with working on a research or industrial project where they apply all of their knowledge from their previous lectures. This may or may not be a team project.

This educational model has its strengths and has served us well but it also has many drawbacks. The students have to sit through at least three years of lectures before they ever really get to be engineers. What we have to realize is that engineers are not defined by their course credit hours and their GPAs; rather they are defined by their ability to innovate and solve practical problems. On day one of university, students have the potential to solve complex world problems and we should let them. Approaching the subject matter from disjoint building blocks is not necessarily the way to hook students and get them excited about the major. Richard Miller of Olin college often draws an analogy between engineering students and violin students. Can you imagine not playing the violin until your fourth year of study? Violinists start making sounds with their instrument the first day of lessons. Why do we create an environment that makes it difficult for students to realize what an engineer actually is? Students are capable of creating and engineering from day one and we should nurture and promote that. Instead of weeding away the students, we should build their capacity and love for engineering. We should make their entire experience (especially the first year!) full of fun and exciting projects, that leave them begging for the knowledge and tools we usually have to force upon them. Many of the students in a traditional engineering curriculum only come to know and understand the essence of engineering by doing extracurricular projects. For example, my involvement with national engineering competitions early in my undergraduate studies was the trigger for me to care about what was being taught in class. Why don't we bring this into the classroom from the start and for *all* students?

We as engineers are often lectured at with little interaction in class, little discussion, little team work, and little connection to the "real" world. In many cases, this is very unfortunate because the world, whether industry or academia, is a collaborative environment that offers a multitude of learning avenues. In the "real" world we discuss, we work together, we solve real problems. Why not make the classroom the real world, instead of pretending it is separate? There is no reason students cannot tackle actual problems early in their education. Few people can make extraordinary impacts in engineering on their own, but many groups of people make great impacts each and every day. I am a strong advocate for collaborative interactive classrooms and I work hard to create that setting. I also will hard to start students early solving actual problems sourced from research and industry.

So far, my teaching experience is based on:

- teaching a large undergraduate engineering course at UC Davis
- the four teaching assistantships I've had in the UC Davis MAE department,
- my work mentoring 15+ graduate and undergraduate students in a lab setting
- supervising and teaching students in the UC Davis engineering machine shop
- leading workshop series at the Davis Bike Collective on bicycle repair over 7 years
- leading action research teams for the Education for UCD Sustainable Living course

Through these I have been exposed to a variety conventional and non-conventional teaching paradigms built on collaborative education, high standards, and engaging students with a variety of learning styles and backgrounds. For example, my teaching through the Davis Bike Collective has built my aptitude for individualized hands off teaching for 10-20 students at a time while each student is working on their individual project. I can make use of combinations of kinesthetic, auditory, visual, and example based teaching and rapidly assess the student's intake and adjust teaching styles as needed. Since my resume details these efforts, I'll only further touch on the most recent course I have taught to demonstrate my classroom teaching experience.

I have taught an introductory class on three dimensional thinking, engineering graphics, design, and computer-aided design both as the lead lecturer and as a teaching assistant. The course materials and details can be found on the course website.

This was a class of about 120 students; most were freshman and the course is promoted as a good introductory class to mechanical engineering. I organized the class much as it was previously taught by another professor, with two hours of lecture per week and a three-hour 30 person laboratory led by four TAs. I was responsible for creating the curriculum for both the class and lab. I encouraged TAs to work together to develop teaching materials for lab sessions around the curriculum design. It is difficult to have a very interactive classroom with 100+ students, but I made use of regular short group assignments during class to have students engage with and learn from their neighbors. I also used a combination of visual materials and in-class sketching examples to explain concepts. Students were assigned weekly readings and sketching along with homework assignments to develop their mastery of that week's material. The lab and lectures functioned somewhat independently, but were designed to reinforce each other. The most difficult process was teaching students how to think visually, especially if they hadn't done so before. Lots of office hour time was spent working with students that had difficulties. I also kept a strong focus on drawing and sketching in the lecture sections, as I still see it as a fundamental skill for design. But the labs focused much more strongly on computer aided visualization and design. Sketching and drawing by hand remains one of the most effective ways of transferring and formalizing ideas and shapes from ones mind. Some of the other teaching methods and tools I implemented and used are:

- Developing three-dimensional thinking through sketching.
- Administered two comprehensive in-class exams.
- In-class small-group problem solving assignments (sketching).
- All lecture notes were posted online after each lecture with reuse permissions where possible.
- I developed a good portion of the visual materials myself, borrowed the rest from permissive online resources, and filled in the rest from the class text book.
- I developed an extensive course website with syllabus, tutorials, and extra class-related info.
- Use of the Sakai (http://www.sakaiproject.org/) UCD software implementation to interact with and share materials with the students. This included email communication and chat room communication.
- Online assignment submission and grading to reduce dealing with paper.
- Regular office hours sessions for small group learning and appointments for individual meetings.
- I worked closely with the TAs to help them develop their lab sessions and teaching skills with weekly feedback to keep track of student progress.

My student reviews of the class revealed that most recommended the class and many found the class to be a positive reinforcement of their choice of major. 85% of the students felt the course and my teaching were good overall. My major weaknesses were the over use of slide shows and the ability to make the lectures fun.

If I were to teach the course again I would change these aspects:

- I'd incorporate 3D printing in the class and create hand held objects for each student to touch and see for explanation of 3D concepts.
- Move away from slide shows by creating large props and designing more 3D interactive computer examples to explain concepts.

- Begin the development of my own text book material.
- Devote more class time to collaborative small-group learning and exercises, leaving more of the lecture material for out of class reading or lecture videos.

The final sections below detail some other aspects of teaching that I plan to incorporate and include some class designs for undergraduates and graduates.

Collaborative Teaching

I am interested in collaborative teaching efforts, as a good team of teachers will often complement each other and exceed what is provided by a singleton. Yes, teachers all have their individual styles but this is no reason not to harness the resources from many teachers to teach the same course while developing plans, materials, etc. I would like to develop and foster collaborative teaching groups in courses that span not only our small departments, but across other colleges around the world.

Grades

Individual performance and scholastic aptitude are certainly a way to evaluate someone's potential, but traditional grading derived from tests and assignments fails to measure many of the other qualities that make successful engineers. The undergraduate experience is highly focused on individual grades. The successful students, in terms of our grade measure, are the ones who figure out how to optimize their behavior for obtaining the highest grade. This optimization does not necessarily include the aspects of engineering that we really want to get across to students. Where possible, I'm for more qualitative measures of a student's success and would not complain at all if traditional grades could be left out of the picture altogether.

Technology

Expanding teaching techniques and the freedom to be experimental are high priorities for me. Students continue to absorb and learn in new ways faster that we can keep up with. One of importance is through modern technology. I am an advocate of utilizing technology in and out of the classroom for collaboration and interactive learning. These methods range from social media and collaborative software to class polling and open accessible content. I plan to make strong use of these latest tools in the classroom, including:

- clicker feedback in classroom
- interactive websites
- maintaing a class blog
- exploration of massive open online courses (MOOCS)
- video podcasts
- class wikis

Learning to teach

PhD students in engineering are typically not taught how to teach. We are measured on our research and scholastic capabilities but not on our teaching abilities, so our time is spent learning how to do the former rather than the later. This is quite unfortunate because many of us are expected to be good teachers in our post graduate careers. We've all had horrible teachers in our undergraduate curriculum and these are a product of a broken system. This means that if we want to be great or even just good teachers we have a lot to learn and we must do this on the job and through extra-curricular learning opportunities. Workshops, readings, collaborative teaching, and mentor-mentee relationships will be a core part of my continuing education on education to make up for the gaps in my skill set.

Text books

Expensive text books in which a new version is published every year will become a thing of the past. The average semester/quarter cost of engineering text books for a student can range from $300-$600. The publishing industry has made this whole system a scam for both the students and the authors, with the publishers reaping all the benefits. This basic knowledge that undergraduates learn, should be available in an open and accessible form for no cost. Wikipedia has shown us the power of the masses to create and maintain the most authoritative text on the world's subjects. This should be no different for fundamental text books. The classes I teach will be accompanied by a collaboratively created text book that will morph and change throughout the years. The materials will be created by both the teachers and the students and will not be limited by what can be put on paper, but will take advantage of the power of the world wide web and the latest internet technologies. Successful examples of things of this nature include UC Davis's ChemWiki, NanoHub, Connexions, and WikiBooks.

Project Based Learning

Young students have a deep reservoir for passion and we as teachers should help that flourish as early in the engineering curriculum as possible. One way to do this is to give students real-life problems to solve and have them use the power of team work to create and innovate solutions.

Course Ideas

I am interested in teaching many classes. I am well equipped to teach undergraduate courses in dynamics, control, bio-mechanics, vehicle dynamics, mixed system dynamics with bond graphs, and aircraft dynamics but am adaptable and open to any courses in a general mechanical engineering curriculum. I also have a personal interest in developing undergraduate programming and software development skills at a early stage. I believe that most engineering curricula gloss over these skills, leaving students to flounder and develop extremely poor programming practices, even though engineers are called on to analyze data and to be experts in scientific computing. An early start at good practices in programming will save large amounts of time in the future and set students up to be marketable for the web generation.

As far a graduate classes are concerned, I am interested in teaching advanced multi-body dynamics, in particular developing students' knowledge of Kane's method and using symbolic computational software for efficient derivation of symbolic equations of motion. I have been involved in the development of software to do this and have plans to engage students in software design and development in this area. I believe that the engineering curriculum is often weak in the development of statistical analyses for experiments. I would like to teach a course on design of experiments that will give graduate students the tools and know-how to work with the massive amounts of experimental data that inevitably carries with it much error. I would also like to teach a graduate level version of software design for science. Big data is at our door step and engineers will need to be able to deal with big data to solve the next generation of engineering problems. Good system identification courses are often hard to come by. I'd like to develop one that is founded and linked to machine learning and big data problems.

Lastly, I am interested in developing a course modeled after a colleague in Spain that centers on the bicycle. The bicycle is an interestingly rich dynamics and control problem and is a familiar object to most people. The class would be designed to emulate the scientific process, essentially provided a canned, but complex, dynamic system to model, measure, and validate with experimentation. The bicycle offers an economical and tractable platform for learning the whole picture in experimentation which is perfect for a undergraduate or graduate level course.

Reproducible Scientific Computing

Prerequisites: undergraduate course in computer programming

Recommended: undergraduate statistics

Suggested level: advanced undergraduates or graduate students

This class will introduce students to modern scientific data analysis which is structured around reproducible computational workflows. Large amounts of data are often overwhelming and managing that data becomes cumbersome and error prone with commonly learned computational tools.Topics will include version control, databases, unit testing, reproducible software, the unix shell, Python, R, BASH, regular expressions, data types, object oriented programming, Make, matrix programming, scientific document creation, and sharing data through the web. The class will also delve into general topics in Open Science and what role massive sharing of data, code, and literature will play in the future. Students will come away with a broad toolset for managing data and software at the research level along with new ideas related to open science. Students will be responsible for creating a complete computationally reproducible project based on real data and sharing it via the web.

Multibody Dynamics

Prerequisite: undergraduate Newtonian dynamics
Recommended: graduate advanced dynamics
Suggested Level: graduate students

This class will explore the derivation of nonlinear and linear equations of motion for constrained multibody systems by utilizing Kane's method. The basic topics include coupled rigid-body kinematics/dynamics, reference frames, vector differentiation, configuration and motion constraints, holonomicity, generalized speeds, partial velocities, mass, inertia tensor/theorems, angular momentum, generalized forces, comparison of Newton/Euler, Lagrange, Kane, Featherstone, etc methods, and orientation: Euler, Quaternion, and Rodrigues parameters. The theory will be tightly coupled with computer-aided symbolic equation derivation and software tools to ease the workflow of complex dynamics problems from derivation to simulation. Groups will be responsible for designing an analytical study of a multibody system with the techniques taught in the course.

Experimental Methods in Biomechanics

Prerequisite: undergraduate Newtonian dynamics, biomechanics theory, computer
programming
Suggested Level: graduate students

This course will familiarize students with modern experimental techniques for biomechanical analysis of human movement. The lab based class will cover data acquisition, force platform analysis, planar and three-dimensional motion capture, inertial measurement units, image processing, data reduction and smoothing, body segment parameter estimation, gait, electromyography, and biomechanical computational analysis. Groups will be responsible for the design of and performing a mini-experiment with the newly learned techniques.

Vehicle Dynamics

Prerequisite: advanced dynamics, control systems, computer programming
Suggested Level: upper level undergrads and graduate students

This course will cover the general theories of vehicle dynamics. We will study the derivation and analysis of the equations of motion for automobiles, trailers, planes, single track vehicles, etc. Topics will include lateral dynamics, stability, control, simulation, visualization, longitudinal dynamics (acceleration and braking), suspension, etc. Demonstrations of real vehicles will be held. Students will be responsible for a completing a group project in the analysis of a vehicle.

Experimental Methods in Vehicle Dynamics

Prerequisite: undergraduate Newtonian dynamics, vehicle dynamics theory, computer

programming
Suggested Level: graduate students

This course will familiarize students with modern experimental techniques for dynamic measurement and analysis of vehicles (automobiles, trains, planes, boats, etc). The lab based class will cover data acquisition, force measurement, kinematic measurements (angular rate, acceleration, GPS, IMUs, etc), data reduction and smoothing, system identification, mass and inertia estimation, and dynamical computational analysis. Each class member will be responsible for a part an class wide team project to measure and validate the dynamics of a common bicycle.

System Identification

Prerequisites: linear algebra, control systems
Suggested Level: upper level undergraduates and graduate students

This course will cover linear and nonlinear system identification. Topics will include discrete systems, data quality, stochastic processes, noise, black box identification, grey box identification, non-linear identification. The course will approach the theory from a practical point of view and students will be responsible for identifying models based on real data.