UnLecture: A Novel Active Learning Based Pedagogical Strategy for Engineering Courses

Vignesh Subbian, University of Cincinnati

Vignesh Subbian is an instructor/teaching assistant in the Department of Electrical Engineering and Computing Systems at the University of Cincinnati. His research interests include embedded computing systems, medical device design and development, point-of-care technologies for neurological care, and engineering education.

Dr. Carla C. Purdy, University of Cincinnati

Carla Purdy is an associate professor in the School of Electrical Engineering and Computing Systems, College of Engineering and Applied Science, at the University of Cincinnati and an affiliate faculty member in UC's Department of Women's, Gender, and Sexuality Studies. She received her Ph.D. in Mathematics from the University of Illinois in 1975 and her PhD. in Computer Science from Texas A&M University in 1986. She is the head of UC's B.S. in Computer Engineering Program and the coordinator of the Preparing Future Faculty in Engineering Program. Her research interests include embedded systems and VLSI, intelligent embedded systems, software and systems engineering, computational biology and synthetic biology, agent based modeling and simulation, mentoring, and diversity in science and engineering.

Dr. Fred Richard Beyette Jr., University of Cincinnati

Fred R. Beyette Jr. has been teaching and performing research in areas related to Mixed Technology Embedded Systems since 1988. From 1988 to 1995, his efforts contributed to the work of materials, device and systems specialists at the NSF Engineering Research Center for Optical Computing Systems. After receiving his Ph.D. in Electrical Engineering from Colorado State University in 1995, Dr. Beyette spent the 1995/96 academic year as a NSF International Postdoctoral Fellow working at the University of Sheffield in the United Kingdom. While in the UK, he investigated the system level feasibility of photonic devices based on piezoelectric multiple quantum well structures which exhibit blue shifting absorption characteristics. In 1996, Dr. Beyette joined the Department of Electrical and Computer Engineering & Computer Science at the University of Cincinnati. He is currently performing research and teaching in areas related to design and implementation of Mixed Technology Embedded Systems His research interests include development of Point-of-Care medical diagnostic systems, Hardware for Wearable Computing Application, Embedded Systems for Smart Power Grid Infrastructure, hardware development of photonic information processing systems, components that bridge the photonic/electronic interface.

UnLecture: A Novel Active Learning Based Pedagogical Strategy for Engineering Courses

Introduction

Our notion of adding participant-driven activities to engineering classrooms stems from the concept of "unconference"[1] in professional meetings and conventions. In recent years, several technical conferences include an *unconference* event in which all conference attendees are invited to join an open-ended discussion on existing and new problems in the field. Any participant can initiate discussion based on the conference/discussion theme or their own research experiences, and other participants can question and/or add to the same topic. This is contrary to a typical conference event where selected speakers/experts are invited to present in a structured track format, hence the name, *unconference*. Professional societies in technology and engineering fields generally conduct an *unconference* in their annual meetings to receive "holistic" views on problems in the field in order to develop research questions and grand challenges. The concept of *unconference* motivated us to develop an active-learning technique called *UnLecture*, to integrate real-world experiences into engineering classrooms. Undergraduate engineering students at the University of Cincinnati (UC) are required to participate in a mandatory cooperative education (co-op) program in which students supplement their academic coursework with professional work experience in their field of study. During or immediately after their sophomore year, students alternate between school and work, completing five co-op rotations (approximately 20 months) in industry and/or research positions. Analogous to research or professional experience of attendees in an *unconference*, students in an *UnLecture* session will have co-op or internship experience that can be used to promote learning and knowledge sharing and also to inform curriculum development and teaching.

Literature Review

One of the core elements of active learning is introducing activity into the traditional lecture-based classroom[2]. Activities such as short in-class discussions[3] and pause procedures[4] are predominantly embedded within lectures. Specifically, discussions in class (at "logical" breaks during lectures) have been shown to improve retention of material and to help students develop thinking skills[5, 6]. The premise of *UnLecture* is, however, based on designing inquiry-based stand-alone discussion sessions that allow students to create connections to both classroom lectures and real-world experiences (see Figure I, Instructional Model of *UnLecture*). This technique is entirely different from round-table discussions[7] that involve analysis and critique of case studies and/or research articles. It also differs from student presentations, since it emphasizes discussion and sharing of professional experience rather than dissemination of information provided by a single student or group of students. Rather, *UnLectures* are based on promoting reflective learning through peer instruction. Studies have shown that reflection of students' own or others' experiences results in development of new perspectives or clarification of concepts and techniques[8, 9]. It is also evident from these studies that reflective learning has significant value in professional practice[10]. Given that our students have integrated cooperative education into their curriculum, *UnLectures* provide meaningful ways to reflect on lessons from both engineering practice and classroom education.

Development of *UnLecture*

The *UnLecture* technique was first piloted in a senior/graduate-level course, EECE 6038C–Advanced Microsystem Design, in Spring 2013. This course is focused on advanced skills for microcontroller-based embedded systems[11]. The pilot *UnLecture* session was based on an open-ended theme: the discipline of embedded systems and technology shift. One of the goals of this session was to experiment with different inquiry-based strategies to enable formal development of an instructional model of *UnLecture*. Reading material[12, 13] was provided to students one week prior to the session. This reading assignment was meant to help interested students gather information needed for discussion or logical arguments. During the session, active-learning strategies such as reaction to a video related to computing education, and inquiry using publication trends in the field of embedded systems, were used to stimulate discussion. Students presented interesting viewpoints on several topics, including but not limited to, hardware for personalized learning in electrical and computer engineering, mobile/ubiquitous computing, and System-on-Chip (SoC) technologies. Two key observations were made during this pilot session: (1) it was evident that students put forth diverse perspectives based on their experiences from cooperative education, and that it is important to utilize student experiences to promote classroom learning, and (2) structured inquiry, as opposed to open inquiry, was more beneficial in helping students relate concepts learned in the course to both their professional experience and the discussion itself. Based on these preliminary results, the technique was formally modeled and deployed in an undergraduate software engineering course during Summer 2013. Additional information and results from the pilot session are presented in a later section.

Instructional Model of *UnLecture*

The *UnLecture* technique is built on a themed, participant-driven discussion session along with reflective writing components before and after the session. The central element that facilitates both the writing and active-learning components is the *UnLecture* rubric. The rubric is a set of carefully designed questions based on the discussion theme, usually provided to students a week before the session. It should be noted that the *UnLecture* rubric is not necessarily a grading rubric. It is rather intended to serve as a “blueprint” to define learning outcomes and guide students and instructors in executing activities involved in a session. The instructional model of *UnLecture,* as shown in Figure 1, consists of three phases: Retrospection, Examination, and Reflection.

- Before the session, students *retrospect* their past co-op/internship assignments, recollect details that are related to the session theme, and document some fine points based on the questions in the rubric.

- During the session, students share their retrospective thoughts and learn from fellow students' cooperative education experiences. They also *examine* practices that were realized in various course projects and assignments, and analyze the differences and similarities between their experiences in industry and their learning experience from the course.
- After the session, the students combine their perspectives from both retrospection and examination to *reflect* on how they will perform differently in their next co-op rotation or work assignment.

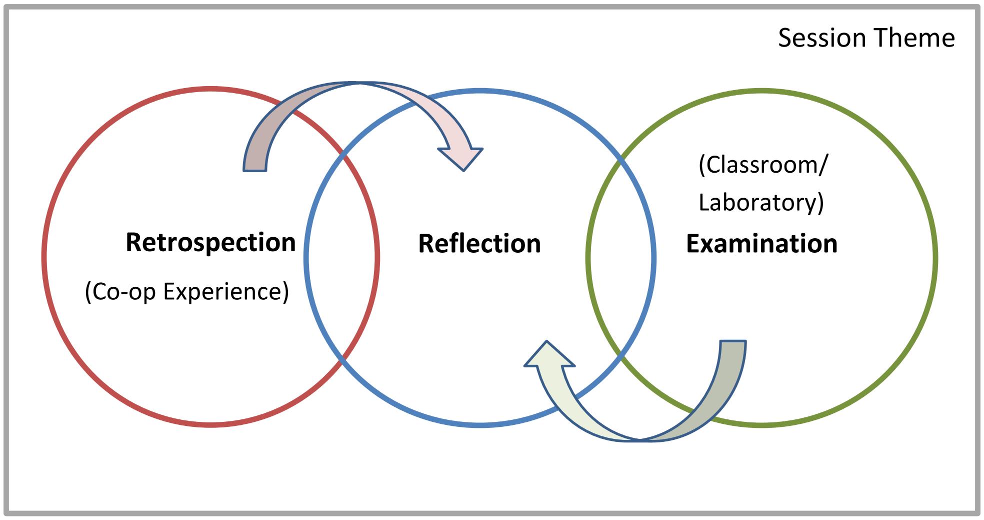

Figure 1 Instructional Model of *UnLecture*

Methods

Implementation of *UnLecture* in an engineering classroom requires meticulous assessment of several aspects of the course. First, class time and student workload for existing traditional lecture modules and other course components such as laboratory projects need to be re-evaluated in order to make time for the *UnLecture* activities. This is important because *UnLecture* requires a reasonable amount of time and work, both inside and outside of the classroom. An *UnLecture* related to a certain topic is typically held after that topic is covered in a traditional lecture or reading assignment. Second, students' background and co-op/work experience information should be collected early in the semester to ensure that every student benefits from *UnLecture*-related activities. If needed, small groups may be formed based on students' co-op experiences so that every group is balanced in terms of technical proficiency and co-op/work experience. Last, it is recommended that both attendance/participation in sessions and the reflective writing portions be given separate weights in the grading system.

Rubric Design

This section presents an example of developing *UnLecture* rubrics for a senior/graduate-level course, EECE 6017C – Embedded Systems. In this course, students learn principles, process models, and practical methods for efficiently developing embedded systems, from requirements gathering and specification through design, implementation, and testing[14]. The laboratory component of this course focuses on rapid prototyping of digital systems using FPGA and high-level design tools. This course is currently being considered for the possible addition of *UnLectures.* As a part of this effort, a trial *UnLecture* was conducted during the Fall 2013 semester. The rubric for this session is shown in Table 1. The proposed grading scheme is 15% of the course grade for a total of four *UnLectures* (5% for attendance and participation in sessions and 10% for the reflective writing component). The weights may be altered based on the number of *UnLectures* included in the course. While only one *UnLecture* was organized, a total of four sessions are proposed for the next offering of this course. The theme and objectives of the planned sessions are as follows:

1. *Application Domains and Process Models*: The first *UnLecture* is typically used to allow students to introduce their co-op job assignments and their responsibilities. Additionally, the focus of this session will be to discuss various application domains of embedded systems. The rubric for this session is the same as the example rubric shown in Table I. It will be slightly revised to include inquiry on process models in the embedded systems industry.

2. *Performance Analysis and Code Optimization*: In the *retrospective* part of this session, students will present their experience in embedded software optimization with respect to one of the following parameters: code size, code efficiency (speed), memory, power. Performance analysis techniques such as profiling will also be discussed. In the *examination* phase, students will revisit their laboratory assignment on code optimization, and provide examples of parts of code that needed optimization.

3. *Reliability, Safety and Security in Embedded Systems*: For this *UnLecture*, students with co-op experience in teams where safety, security, and/or dependability are major design requirements will be identified and assigned as moderators. The objective of this session is to review and emphasize concepts that are important to developing safety-critical and secure embedded systems.

4. *Embedded Systems Research*: This is a non-traditional *UnLecture* where the focus is on academic research experience rather than industry experience. Undergraduate students will be teamed with graduate students for this exercise, and the graduate students will serve as session moderators. In the *retrospective* phase, the emphasis will be on topics such as developing a research hypothesis, experimental methods, industry jobs versus research careers, and examples of graduate research projects in embedded systems. In the *examination* phase, students will present examples of how classroom learning aids in identifying and/or solving their research problems. The purpose of this session is to help students broaden their views on graduate studies, especially since the majority of undergraduate students are either offered jobs by their co-op employers or hired by other industry employers based on their co-op record and hence may not seriously consider continuing on to graduate research.

Table I Example of *UnLecture* Rubric

Prelude: In this section, briefly describe your past co-op/internship/work experiences that you think are most relevant to this course or embedded systems, in general. Include company name, summary of your responsibilities, duration of each assigned task, and duration of the project.

Retrospection: For each project listed in the prelude, write a retrospective essay based on the following questions:

1.1. There are different abstraction levels (and sub-levels) in embedded system design: hardware (architecture, microarchitecture, circuit-level, gate-level, RTL), software (system software/firmware, OS/RTOS, application code).
  - List the abstraction layer for each project listed in the prelude.
  - What challenges or issues have you encountered in dealing with these various layers?

1.2. Different application domains (of embedded systems) have different design philosophies and different tool chains and require different skill sets.
  - Share some of your application-specific knowledge gained through co-op/work experience. In your own words, explain the connection between application needs and embedded hardware.
  - What tools/programming languages did you use, and how did they help accomplish tasks?

1.3. Briefly explain system-design activities of the team you were involved in.

Examination:

2.1. What are your thoughts on the design choices in the laboratory project of this course? How is the class project different from or similar to your past experiences (include pros and cons)?

2.2. Give specific examples of how the knowledge gained in industry was applied in this course.

Reflection:

3.1. What did you learn from this *UnLecture* session? In other words, what are your thoughts on conversations that you had with your classmates and their experiences?

3.2. Based on this discussion, what will you do differently in your next co-op/industry work assignment?

Results

EECE 6017C-Embedded Systems (Fall 2013): A total of 35 students (20 undergraduate, 15 graduate) were enrolled in this course. The enrollment demographics based on majors is shown in Figure 2. Out of 21 students with co-op or industry experience, 10 students had work experience in embedded systems. Table II shows *UnLecture* evaluation results. It was evident from the feedback that students wanted more *UnLectures* to continue the conversation in more depth. The first discussion session, however, is usually meant to the set the tone of *UnLecture*, and conversations are expected to be superficial. Also, peer instruction may not be obvious in the very first *UnLecture* session, but it has been observed to set in during the second or third sessions, in courses where *UnLectures* have been completely integrated. The following are a few excerpts of student feedback:

- "Many of my peers had worked in small companies doing board layouts and firmware development. I would like to know more details on their work because it is entirely different [from] my co-op."

- “I learned differences in design process models between startups, medium, and large-scale companies”
- “I would like to have further discussions on design and testing strategies…”
- “It was interesting to hear expectations of the embedded systems industry…and what skills and knowledge areas they are looking for…”

Table II *UnLecture* Evaluation Results (Fall 2013)

Course-specific Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
The *UnLecture* sessions helped me understand the “climate” of the embedded systems industry.	0%	0%	3.3%	66.7%	30%
I was able to engage and learn from my peers during this session	0%	16.7%	30%	46.6%	6.7%
I would like to see more *UnLectures* in this course	0%	0%	6.7%	33.3%	60%
I was able to relate my industry experience to the education/curriculum at U__ (excluding first-year/new graduate students)	0%	0%	3.3%	70%	26.7%

EECE 3093C-Software Engineering (Summer 2013): This is the first and only course to have *UnLectures* completely integrated into classroom instruction[15]. Five *Un Lectures* were deployed and tested during Summer 2013. Complete rubrics and assessment results are also available[16]. With minor changes, some of the rubrics developed for this course can be reused in other courses. The software testing rubric for this course, for example, can be easily modified to deliver a similar *UnLecture* in the embedded systems class.

EECE 6038C-Advanced Microsystem Design (Pilot UnLecture; Spring 2013): This course consisted of 35 students (24 undergraduate, 11 graduate) from three majors (see Figure 2). Twenty-seven students had co-op or other industry experience. As the first audience to experience *UnLecture,* this class was instrumental in leading to the concept of *UnLecture* rubrics. The following are some excerpts from student feedback:

- “I wish we had covered more advanced material earlier (in the course). May be more case studies like the unlecture.”
- “Expand more on file management and have more of those round-table discussions, and give us more opportunities to talk in those discussions. Great course, learned a lot!”
- “Closer examination of memory usage and resource cost/performance analysis.” (This comment refers to content delivered as a lecture, but it is also an example of a topic that can be examined in an *UnLecture*.)

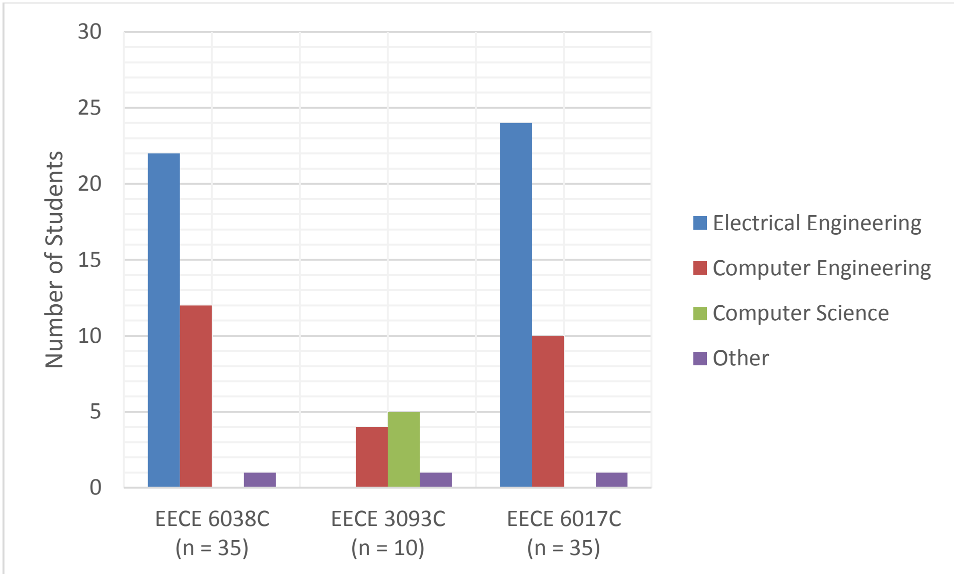

Figure 2 Enrollment Demographics of Courses with *UnLecture*

Discussion

This paper has mainly described the structure and methods of the *UnLecture* technique, along with preliminary results that support the feasibility of this technique in engineering classrooms. While it specifically addresses how *UnLectures* can be applied to courses in the embedded systems curriculum at UC, the technique is also applicable to several other electrical and computer engineering courses and possibly to courses in other engineering disciplines as well. Furthermore, it is worth pointing out that senior capstone design courses in any engineering discipline are an ideal venue for implementing *UnLectures.* At UC and several other institutions with co-op programs, in particular, undergraduate students in senior design classes would have completed all their core academic coursework and 20 months of cooperative education in industry. This is seen as a perfect opportunity to effectively utilize *UnLectures* to reflect on the connection between engineering education and engineering practice.

References

1. M. Nandeesha, "Chapter 6 The rise of unconferences for professional networking and knowledge sharing: a case study of "libcampbangalore," *Best Practices for Corporate Libraries*, pp. 99, 2011.

2. M. Prince, “Does active learning work? A review of the research,” *Journal of Engineering Education*, vol. 93, no. 3, 2004, pp. 223-231.

3. C. C. Bonwell and J. A. Eison, “Active learning: Creating excitement in the Classroom,” *ASHE-ERIC Higher Education Reports*, ERIC Clearinghouse on Higher Education, 1991.

4. K. L. Ruhl, C. A. Hughes, and P. J. Schloss, “Using the pause procedure to enhance lecture recall,” *Teacher Education and Special Education: The Journal of the Teacher Education Division of the Council for Exceptional Children*, vol. 10, no. 1, 1987, pp. 14-18.

5. W. J. McKeachie, “Research on college teaching: A Review,” *ERIC*, 1970.

6. W. J. McKeachie, “Research on College Teaching: The historical background,” *Journal of Educational Psychology*, vol. 82, no. 2, 1990, pp. 189.

7. T. Kunberger and R. O'Neill, “Engineers of the round table: Utilizing a discussion forum to enhance student learning in geotechnical engineering,” In *2010 Annual Conference and Exposition, American Society for Engineering Education*, 2010.

8. E. M. Boyd and A. W. Fales, “Reflective learning key to learning from experience,” *Journal of Humanistic Psychology*, vol. 23, no. 2, 1983, pp. 99-117.

9. C. R. Baker, “Reflective learning: A teaching strategy for critical thinking,” *The Journal of nursing education*, vol. 35, no. 1, 1996, pp. 19-22.

10. K. Thorpe, “Reflective learning journals: From concept to practice,” *Reflective Practice*, vol. 5, no. 3, 2004, pp. 327-343.

11. V. Subbian and F. R. Beyette, “Developing a new advanced microcontrollers course as a part of embedded systems curriculum,” *IEEE/ASEE Frontiers in Education Conference*, 2013, pp. 1462-1464.

12. T. A. Henzinger and J. Sifakis, “The discipline of embedded systems design,” *Computer*, vol. 40, pp. 32-40, 2007.

13. P. Caspi, A. Sangiovanni-Vincentelli, L. Almeida, A. Benveniste, B. Bouyssounouse, G. Buttazzo, I. Crnkovic, W. Damm, J. Engblom and G. Folher, “Guidelines for a graduate curriculum on embedded software and systems,” *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 4, pp. 587-611, 2005.

14. V. Subbian and C. Purdy, “Redesigning an advanced embedded systems course: A step towards interdisciplinary engineering education,” *IEEE Integrated STEM Education Conference*, 2013.

15. V. Subbian and C. Purdy, “A hybrid design methodology for an introductory software engineering course with integrated mobile application development,” *ASEE Annual Conference*, 2014.

16. V. Subbian and C. Purdy, “UnLecture: Bridging the gap between computing education and software engineering practice,” *ASEE Annual Conference*, 2014.