Reengineering the Curriculum: Design and Analysis of a New Undergraduate Electrical and Computer Engineering Degree at Carnegie Mellon University

STEPHEN W. DIRECTOR, FELLOW, IEEE, PRADEEP K. KHOSLA, FELLOW, IEEE,
RONALD A. ROHRER, FELLOW, IEEE, AND ROB A. RUTENBAR, SENIOR MEMBER, IEEE

Invited Paper

In the Fall of 1991, after approximately two years of development, the department of Electrical and Computer Engineering (ECE) at Carnegie Mellon University (CMU) implemented a new curriculum that differed radically from its predecessor. Key features of this curriculum include: Engineering in the Freshman year, a small core of required classes, area requirements in place of most specific course requirements, mandated breadth, depth, design, and coverage across ECE technical areas, a relatively large fraction of free electives, and a single integrated Bachelor of Science degree in Electrical and Computer Engineering. In this paper we review the design of this curriculum, including a taxonomy of problems we needed to address, and a set of general principles we evolved to address them. The new curriculum is described in detail, including new data from an ongoing analysis of its impact on students' curricula choices.

I. Introduction

Current engineering practice has, by necessity, evolved to keep pace with technology: witness the rate at which fundamentally new ideas are introduced into new products. One might suppose, then, that current engineering education has also evolved to track such new developments. However, we argue that engineering education has really evolved only to the extent that individual engineering courses have been updated—usually with increased density of content—to reflect new developments. The prevailing philosophy of engineering education—teach first the basics in mathematics and science, follow with exposition of engineering applications—has remained unchanged and unchallenged for more than four decades. While contributing to the creation of engineers who are current in specific technologies, we believe that teaching of unmotivated math and science followed by incrementally updated technical courses is fundamentally flawed. It contributes little to the education of engineers who can acquire new knowledge as necessary, cope with dynamically changing work environments, or excel in nontraditional jobs. We believe that real impact in engineering education will be made only by looking at the curriculum *as a whole*, in the context of present technological and societal needs, and not just by constant repolishing of aging courses. It is not our intention to imply that engineering education has completely failed in its goals. Rather, we wish to drive home the point that there are advantages to be found in taking a fresh, *unfettered* look at the undergraduate curriculum.

Of course, curricula have tremendous inertia, and often resist all but the most incremental and cosmetic of changes. Unfortunately, many of the problems faced by engineering educators are not amenable to simple, incremental fixes. In October 1989, the college of engineering at Carnegie Mellon University (CMU) instituted a review process across all engineering departments. The goal was to evaluate how well the educational mission of the college was being conducted, with an eye toward redefining both collegewide and department-specific curriculum requirements. Because of the breadth of this undertaking, each engineering department was allowed to consider the best possible curriculum changes, not merely those that could be wedged conveniently into its current web of requirements, prerequisites, constraints, and customs. This paper describes the design and implementation of the new Electrical and Computer Engineering Bachelor's degree program that emerged from this process. This curriculum, which took approximately two years to design fully, was implemented within the de-

Manuscript received December 7, 1994; revised April 12, 1995.
The authors are with the College of Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 USA.
IEEE Log Number 9413199.

partment of Electrical and Computer Engineering (ECE) in the Fall of 1991, and produced its first four-year graduates in the Spring of 1995.

Within ECE, the curriculum was designed by a committee whose quickly adopted name reflected the spirit of process: the *Wipe-the-Slate-Clean Committee*. Composed of eleven faculty from across the breadth of the department's research and teaching areas, the committee interviewed both students and faculty, and worked aggressively for roughly one year to dissect, analyze, disassemble, and finally redefine the ECE undergraduate curriculum. The new curriculum that resulted from this process hinges on a few key ideas:

- Engineering courses begin in the Freshman year, concurrent with mathematics and science.
- The core of required "essential" engineering classes is extremely small.
- Area requirements across a spectrum of electrical and computer engineering topical areas replace most specific course requirements.
- Breadth, depth, and coverage are mandated across this spectrum of technical areas, but individual courses are not prescribed; students flexibly choose from among available topical areas.
- Nearly a full year of the curriculum is unconstrained.
- At completion, the curriculum offers a single, unified Bachelor's degree in Electrical and Computer Engineering.

The end result of our exercise is a curriculum which has been recently reviewed by ABET for accreditation under the ABET "innovative curriculum" clause that permits thoughtful experimental curricula that diverge from existing ABET standards to be considered on their merits. While the final outcome of the accreditation process will not be known until late 1995, comments made by the visiting team were favorable. Also, initial analysis of the three groups of freshman entering ECE in 1991, 1992, and 1993 (about 150 students in each group) indicates that the students are enthusiastic about starting engineering classes in the Freshman year and that these are helpful to the student when selecting their major. There is also evidence to show that the flexibility in the choice of electives has not resulted in a mass exodus to "easier" courses. In general students continue to elect challenging courses to suit their interests.

In this paper we share some of the details of the design process for this new curriculum, and an analysis (ongoing) of its implementation and impact.[1] Of course, we were not alone among universities as we embarked on our reengineering efforts; for example, Drexel, Rose–Hulman, and Texas A&M were already restructuring their curricula as we began our efforts, and as well the US National Science Foundation was organizing Engineering Education Coalitions with similar intent. Nevertheless, we did not join any of these efforts for fear of diluting our own efforts. So rather than attempt a broad survey of competing curriculum strategies, we focus entirely and closely on our own redesign effort, from beginning to end. We offer this as one case study for how one department reengineered its curriculum.

The remainder of the paper is organized as follows. Section II begins by summarizing our motivations for undertaking this effort. Section III offers a taxonomy of the basic problems faced by any electrical or computer engineering department as it struggles to keep pace with technology, students, and society. Section IV describes the design principles for the new Carnegie Mellon ECE curriculum that we evolved in response to these problems. Section V describes the details of the new curriculum, and some of its novel characteristics. Section VI describes its implementation, and recent efforts to analyze its impact on students. Finally, Section VII offers some concluding remarks.

II. Motivations

A. Why Change?

By any traditional measure in 1991, the ECE department was doing well educating its students. The department as a whole was consistently ranked among the country's top EE departments [3] (Components of the graduate program were likewise being ranked highly [4]). The department attracted outstanding undergraduate students: ECE was the first choice among engineering departments of most entering Freshman. Our graduates were recruited heavily by US companies, and the ECE department was on the list of must-visit departments for many companies that recruited only among a select set of elite schools. Our graduates who chose to pursue an advanced degree went on to elite graduate schools.

So, why did we undertake a substantial reorganization of our curriculum? The answers are not simple, nor are they independent. We categorize our broad concerns in the following subsections, beginning with a quick overview of the original ECE curriculum as it stood in the 1990–1991 academic year. These concerns can be regarded as the beginnings of a set of "specifications" for a new curriculum.

B. Original CMU ECE Curriculum

In 1991, the ECE department offered two four-year ABET-accredited Bachelor of Science degrees: the Bachelor of Science in Electrical Engineering (BSEE) and the Bachelor of Science in Computer Engineering (BSCE). Both curricula shared a common Freshman year emphasizing mathematics, science, and computer programming. They also shared a common core of engineering classes, emphasizing linear circuits, electronics, solid state devices, digital logic design, and microprocessors. In addition, these curricula (as did all curricula in the colleges of engineering and science) shared common requirements for humanities and social science courses (called H&SS) that amounted to roughly one such course per semester. An overview of the curricula appears in Table 1.

[1] See [1] for a more detailed, contemporaneous account of this process, and [2] for a more recent review.

Table 1 Original Carnegie Mellon EE and CE Curricula

Electrical Engineering	Courses	Computer Engineering	Courses
Mathematics & Sciences		Mathematics & Sciences	
Calculus	2	Calculus	2
Differential Equations	1	Differential Equations	1
Linear Algebra	1	Linear Algebra	1
Probability	1	Probability	1
Physics	3	Modern Math	1
Chemistry	1	Physics	3
Computer Programming	1	Chemistry	1
		Computer Programming	1
Electrical & Computer Engineering		Electrical & Computer Engineering	
Intro Digital Systems	1	Intro Digital Systems	1
Linear Circuits	1	Linear Circuits	1
Intro Electronic Devices	1	Into Electronic Devices	1
Electromagnetics	2	Computer Architecture	1
Signals & Systems	2	Concurrency & Real Time Systems	1
Analog Circuits	1	Digital Integrated Circuits	1
Digital Integrated Circuits	1	Logic & Processor Design	2
EE Elective	1	Computer Science	
Senior Design Elective	1	Fundamentals of CS	2
		CS Elective	1
Electives		Electives	
Freshman	2	Freshman	2
Engineering Science	2	Engineering Science	2
Technical	5	Technical	5
Free	1	Free	1
Humanities & Social Sciences	8	Humanities & Social Sciences	8

After this common core, the two curricula diverged. The BSEE emphasized traditional electrical engineering topics such as electromagnetics, analog circuits, and signals and systems. The BSCE emphasized computer hardware and software topics such as computer architecture, processor design, data structures, and concurrency. Both curricula required several technical electives, and a capstone design elective.

In 1991, about 40% of our students pursued the BSEE, and about 50% pursued the BSCE. Roughly 10% of our students chose to double major in both electrical engineering (EE) and computer engineering (CE). This was accomplished at the sacrifice of most elective classes: Students completed the core requirements of one curriculum using the elective slots provided in the other. Also, a few of our students double-majored in computer engineering and computer science (which is in a separate college at Carnegie Mellon). This essentially required that all elective classes in the BSCE curriculum were chosen to complete computer science core requirements.

C. Remove Structural Impediments to Accommodate Incremental Change

Curricula usually evolve by accretion, with new requirements and constraints often layered incompatibly on top of existing structures. The resulting rigid course sequences connected by spaghetti-like chains of prerequisites are difficult to modify. This was certainly true of our original EE and CE curricula, and by extension, likely true in many similar Electrical Engineering departments that evolved over the last two decades to become departments of Electrical and Computer Engineering, or departments of Electrical Engineering and Computer Science. In our own case, the end result was that even incremental changes became difficult to implement.

In the original parallel BSEE and BSCE curricula, even modest changes rippled in undesirable ways throughout the two programs. An example makes this concrete. As a result of an ABET accreditation visit, we were asked to add a linear algebra class as a graduation requirement. We responded enthusiastically, on the assumption that we could migrate the course into the early years of the curriculum, and thus make it a prerequisite for our linear circuits class. In this position, it would strengthen the background of all EE students in our circuits and electronics courses, and broaden the background of our CE students by exposing them to more noncalculus mathematics.

Unfortunately, this ideal proved impossible to implement. There was no small-scale alteration of the BSEE and BSCE course sequences that could permit the linear algebra class to be taken by all students before the courses that would use it as a prerequisite. This problem derived from the slight differences in the first few years of the BSEE and BSCE requirements. The BSCE student began to take computer science classes fairly early, so that Junior and Senior computer engineering courses were correctly synchronized with their computer science prerequisites. In contrast, the BSEE student had no such requirements. The end result was that we required our students to take a linear algebra class, but we did essentially nothing to exploit this background in other ECE core classes. This simple example makes clear how difficult it can become to achieve the goal of *uniform* mathematics, science, and engineering core preparation for both BSEE and BSCE students.

D. Rationalize Requirements for Topical Coverage and Workload

As has become amply clear over the last decade, the disciplines of electrical and computer engineering are expanding rapidly as new technical discoveries are made and applied. Likewise, society is placing increasing demands on our graduates to apply their skills in new contexts, and to appreciate and manage intelligently the consequences of their technical decisions. Consequently, the number of "critical" topics to which ECE students could profitably be exposed is also expanding. What is not expanding is the time we have to educate someone to level of the Bachelor's degree.[2] Coming to grips with this accelerating problem was at the heart of our motivation for a significant restructuring of our curriculum.

The original ECE curriculum required a large number of core classes, designed to ensure familiarity with a substantial subset of traditional EE and CE topics. After a great deal of argument and discussion, we came to believe that this approach, which implicitly assumes all students need exposure to (almost) all areas, was no longer credible as the core of a curriculum for the 21st century. Such a strategy mandates that we compress ever more material into the same number of classes. Many of our courses had already fallen victim to "units-creep," i.e., challenging classes meant to require 12 hours of work per week had inflated to require 15 or 18 hours of work from even the best of students. This was caused by well meaning faculty working hard to give students the best, most thorough view of as many topical areas they could—usually with the assumption that this was the only opportunity students would ever have to see the material.

While certainly not opposed to demanding classes, we concluded that the overall strategy of putting more material into the curriculum had become decreasingly effective. Students were being asked to absorb increasing amounts of material, which left less time for reflection, for alternative perspectives on similar technical problems, and for revisiting background material to ensure comprehension. The unpredictable preparation of entering students only exacerbated this problem: we kept discovering that many of our students had never seen material fundamental to the background of our core courses. The end result was that by forcing students to juggle too many topics with too little time to master these topics, many students were learning even less material, less well.

E. Emphasize Engineering Ideas Over Techniques

A related consequence of the explosion of material was that many students came to view their courses as a set of unrelated hurdles to be overcome. As a result, many students were acquiring only a bag of seemingly unrelated problems and solution techniques, without ever really understanding the big ideas that bind and inform these techniques.

Conventional wisdom suggested that after first teaching a vast body of fundamental mathematics and science—which students absorbed like sponges—we were free to teach engineering principles, drawing as necessary on the deep well of basic knowledge internalized by the student in these early studies. This was (and is) a lovely idea, but depressingly unrealistic. Students often had weak or wildly varying preparation in K–12 mathematics and science, and hence uncertain motivation to master the rigorous college level versions of these fundamentals. When a flood of engineering ideas was introduced on top of this precarious foundation, the outcome was often less than satisfactory. Too often, students only had time to focus on the mechanical problem-formula-solution aspects of the topics, without developing a deeper sense of the fundamentals, the interconnections, and the real ideas.

This is especially unfortunate in a fast-moving discipline, where the half-life of a Bachelor's degree is probably less than a decade, and a solid understanding of the "big picture" is the most successful base from which to acquire new skills. As educators, we do our students a disservice if we fail to impart a coherent, connected view of the ideas that define our discipline.

F. Support Interdisciplinary Studies

The most creative and far-reaching contributions are often made by individuals at the boundaries of several disciplines. Likewise, society is placing increasing value on engineers who can apply their skills across disciplines, and can evaluate intelligently the broader consequences of their actions. ECE is an extremely wide field, and many of its most exciting frontiers—very large scale integrated circuits (VLSI), microelectromechanical systems (MEMS), electronic materials, computer-aided manufacturing, telecommunications networks, supercomputing—have strong and established interdisciplinary linkages. However, our original curriculum did little to encourage the creation of engineers who could work comfortably across the boundaries of several disciplines.

The original curriculum implicitly assumed that there were only two sorts of engineers: EE's and CE's. These were produced by completion of a large, rigid core of EE or CE engineering classes. Although industry specifically, and society generally might have valued highly a student who had completed, say, 60% of the EE core classes and 40% of the CE core classes, we had no mechanism for giving this broad individual a degree. Nor did we have any mechanism for coping with an even broader individual who might have wished to complete, say, 30% of the EE core, 30% of the CE core, then a dozen classes in mechanical engineering, operations research and Japanese language, in preparation for a career in computer-aided manufacturing. Indeed, a key conclusion of the early discussion of the Wipe-the-

[2] An alternative is, of course, to extend the amount of time required to educate students to some minimum level of professional competence. Such an approach was advanced by the Massachusetts Institute of Technology in [5] which proposed a five-year accredited Master's program as the principal mechanism for educating entry-level engineers. We return to this idea in Section VI.

Slate-Clean committee was that we would like not only to tolerate such individuals, but to encourage them.

III. Curriculum Design: Problems

A central tenet of any engineering education is that no elegant solution is likely to be found for a problem that lacks a crisp definition. Unfortunately, curricula are complex, often unwieldy creations subject to conflicting demands from the university, from faculty, from students and their parents, and from the industries that employ graduates. Nevertheless, over the course of its deliberations, our committee kept returning to several specific problems which crystallized as the basic issues to address. We summarize these here.

A. Student Preparation is Incomplete

American K–12 education can be blamed for the incomplete mathematics and science preparation of many of our students. Nevertheless, allocating blame does nothing to improve the preparation of our students after they arrive. Moreover, entering students are simply different than they were in past decades: less homogeneous, more diverse in their personal goals and career aspirations. Any curriculum redesign must deal with the following facts:

- Students have less facility and depth in the technical areas we expect all students to have seen, for example, algebra and geometry. Some unremarkable mathematical manipulations that appear frequently in introductory science and engineering classes severely tax many students.
- Students—even the best students—have seemingly random gaps in their backgrounds. In the course of our meetings, the Wipe-the-Slate-Clean Committee talked to a superb Senior EE student, a straight-A student who was being aggressively pursued by elite graduate schools. Yet she mentioned to us that she was very uncomfortable in her first circuits class, *having never seen a complex variable before.*
- Most students have almost no basic laboratory skills when they enter the department, for example: how to keep a lab notebook; how to observe an experiment; how to deal with significant digits and experimental error; how to use orders of magnitude and quick-and-dirty calculations to estimate whether a measured result is in the right ballpark or has gone badly awry, etc.
- A related point: students have virtually no hardware tinkering skills. Previous generations of EE's were notorious tinkerers, with radios and motors and the like. Upon entering college, they knew what a wire was, and a battery. They knew how to solder and read the resistor color code. This is no longer true, and the most elementary of hardware skills—what a wire is, what it does, how it can and cannot connect to a battery—must now be taught explicitly. (This is not exactly surprising, given the inaccessibility of the insides of most electronic products these days.) Our students are now much more likely to have software tinkering experience. However, many students, especially those from less well off high schools, arrive without any exposure to programming ideas or hardware concepts.
- Student expectations and faculty expectations often differ. Roughly speaking, we tend to assume students have the background, energy and motivation to go acquire whatever mathematics, science, lab skills, etc., that they lack, if we send them off in the right direction. (This has always been true of the best students.) In contrast, many students tend to assume that we will teach them *every* topic—the big ideas as well as the basic mechanical skills, the central topics as well as the peripheral background material—without independent initiative on their part.

Any solution here must reconsider how and where in the curriculum to teach these fundamentals, and to what level of detail.

B. Student Perspective on EE, CE, and Subdisciplines is Lacking

By the time they are Seniors, faculty usually expect students to make intelligent choices when they have the opportunity to choose an engineering elective course. Students are expected to ask their faculty advisers for guidance here, and to listen to whatever advice is offered. Our experience as educators suggests that it is already questionable whether this works for Seniors, who have a fairly extensive technical background. However, it is clear that students taking their very first course in a core ECE area like solid state devices or computer architecture are usually not clear about how this area connects to the rest of ECE as a whole.

This was a particular problem in our original 1991 curriculum. At this time, ECE offered two parallel curricula: the BSEE and BSCE tracks, one of which students had to choose sometime during the Sophomore year. The problem was how to educate students to make an informed choice. Certainly, some students arrived absolutely decided on one track or the other. But many relied on our introductory courses to paint a sufficiently broad picture of the discipline for them to make a choice. Unfortunately, these introductory courses concentrated almost entirely on packing in as much engineering material as possible. As faculty, we were often surprised when, after a few weeks in class, in the middle of some intricate technical discussion, a brave Sophomore would ask something like this:

> Exactly what does a computer engineer do? And how does this material help me to be a computer engineer? Is this different from computer science? Is the difference that *we* do hardware and *they* do software? When I graduate will I only be able to design big computers, or do computer engineers do something else as well? And why am I taking all these circuits classes—isn't that for the electrical engineers?

The emphasis on maximizing technical content in those few hours per week left little time to address all these questions satisfactorily. And as the breadth of the discipline continues

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.