If you are like most university professors, you were not taught anything about how to teach in graduate school or when you began in your first faculty position. All you had to go on was how your professors taught, but nobody taught them anything about teaching either. It doesn’t make a lot of sense, but that’s our system.

Teaching is too complex and too important a profession to let people do it with no training or experience. Granted, some new faculty members are excellent the first time they get in front of a class, and I hope you’re one of them, but the odds are against it. There are also a few who are poor teachers from the outset and never get better. Since you’re taking the time now to read a paper about engineering education the chances are that you’re not going to be one of those either. You are probably in the broad middle category of faculty members who have the potential to be excellent teachers but may take years learning how to do it by trial and error. Not that trial-and-error learning is always a bad thing, but in the case of teaching the ones paying the penalty for the errors are not the ones making them. Trial and error is also unnecessary. A lot is known about what makes teaching effective: spending some time in the literature learning about it can knock a couple of years off your learning curve.

The greatest initial barrier to learning new material is often jargon—unfamiliar terms that may denote easily learned concepts but whose unfamiliarity makes them sound esoteric and difficult. If you read ASEE conference proceedings or the *Journal of Engineering Education* or any other teaching-related journal, you will notice that a number of terms keep showing up, often with little or no explanation. If you don’t know what they mean, the articles in which they appear may be difficult to decipher. The purpose of this paper is to help you over this hurdle. The few terms to be defined don’t even begin to constitute an exhaustive glossary of educational jargon, but if you understand them you’ll be off to a good start.

We first introduce *learning styles*—the different ways students characteristically use to take in and process information. Understanding what those ways are is a good first step toward designing instruction that can accommodate the learning needs of all of the students in a class. We then define three instructional approaches: *active learning* (getting students to do things in class that actively engage them with the material being taught), *cooperative learning* (putting students to work in teams under conditions that promote the development of teamwork skills while assuring individual accountability for the entire assignment), and *problem-based learning* and similar approaches (teaching material only after a need to know it has been established in the context of a complex question or problem, which increases the likelihood that the students will absorb and retain it).
Our focus next shifts to planning courses and measuring learning outcomes. We begin by defining and illustrating learning objectives—explicit statements of what students should be able to do when they have completed a segment of a course. A good set of objectives can be an invaluable resource for planning courses and individual lessons, creating assignments and tests, and defining the course in a meaningful way for other faculty members preparing to teach it, instructors of prerequisite and subsequent courses, and accreditation visitors. In the remainder of the paper, we introduce Bloom’s Taxonomy of Educational Objectives, a system for classifying learning objectives according to the skill level required to meet them; define and distinguish the terms assessment and evaluation, two related processes that are vitally important in every aspect of both teaching and research; and discuss ABET and the engineering program accreditation process, in which the quality of every engineering department in the country is periodically assessed and evaluated.

Learning Styles

Students come with a wide variety of abilities, attitudes, interests, ambitions, and levels of motivation, and instructional methods that are effective for some students may be relatively ineffective for others. For example, one engineering student might be comfortable with relatively abstract theories and mathematical models and another might be much more receptive to concrete (“real-world”) material such as lab experiments and industrial plant operations. A theoretical and math-intensive course would probably be much more effective for the first of these students, and a practical hands-on course would be a more positive experience for the second one.

A student’s learning style is the way he or she characteristically takes in and processes information. Learning styles provide good clues to the instructional methods students are most and least comfortable with. If you know the range of styles that categorize the students in your class, you can design balanced instruction so that all students are taught sometimes in the manner they prefer, keeping them from becoming too uncomfortable to learn, and sometimes in their less preferred manner, forcing them to stretch and develop skills in areas that they might be inclined to avoid if given the choice.

Several learning style models have been developed and applied to engineering education. One formulated by Felder and Silverman [1988] involves four dichotomous dimensions. Students may be

- sensing learners (concrete, practical, oriented toward facts and procedures) or intuitive learners (conceptual, innovative, oriented toward theories and meanings).
- visual learners (prefer visual representations of presented material—pictures, diagrams, flow charts, etc.) or verbal learners (prefer written and spoken explanations).
- active learners (tend to learn by trying things out, working with others) or reflective learners (tend to learn by thinking things through, working alone).
- sequential learners (linear, orderly, tend to learn in small incremental steps) or global learners (holistic, systems thinkers, tend to learn in large leaps).
Most engineering instruction in the past few decades has been heavily biased toward intuitive, verbal, reflective, and sequential learners, although relatively few engineering students fall into all four of these categories. The result is that most engineering students are taught in a manner at least partially mismatched to their learning styles, which could hurt their performance and their attitude toward engineering as a curriculum and career.

At <http://www.ncsu.edu/felder-public/Learning_Styles.html> you will find links to papers that provide extensive information on the Felder-Silverman model, including characteristics of students with different styles, teaching methods that address each style, suggestions for achieving the desired balance, and an online instrument to assess preferences on each of the four dimensions of the model. Other papers on the same site provide information on other learning style models and cite references to their applications to engineering education.

**Active Learning**

During a traditional lecture, the only one who is active is the lecturer—talking, writing on the board, showing transparencies, asking questions and often supplying the answers when there is no response from the class. The students are passive—watching and listening and taking notes (maybe), but seldom actively thinking about the material being presented.

Unfortunately, that’s not how people learn. We know from cognitive science that information received passively with no attendant action or reflection is not retained in long-term memory. The cliché about something going in one ear and out the other is a good metaphor for what happens to material presented in traditional lectures. Compounding the problem is that students sitting passively in a lecture invariably take mental breaks in which their minds go elsewhere, and the longer they sit, the more frequently those breaks occur and the longer they last. If you’re thinking about your homework in other courses or your email backlog or how long it still is to lunch, you’re not hearing the lecture, and when you do get back to it, what you missed could make what you’re hearing now incomprehensible. After a while, the lecture is just background noise.

**Active learning** is anything that happens in a class that engages students with the material being presented. Students might be called on to work individually or in small groups for brief periods of time to answer questions, start problem solutions, fill in steps in a problem solution or derivation, brainstorm lists, troubleshoot processes, or think of questions about the material just lectured on. At the end of the allotted period, the instructor calls on several individuals or teams for their responses, then collects more responses from volunteers, and moves on when the correct answer has been obtained and it seems clear that the students understand it.

Good things happen in a class when active learning is used, even if it’s only for a few minutes out of an hour-long class. Activity refocuses students who have drifted off into mental breaks and energizes the entire class. If the activity requires the students to do something they will later have to do on homework and tests (such as draw and label a flowchart or free body diagram, outline the solution of a problem, estimate the value of a process variable, do some computations or parts of derivations, or come up with a theoretical interpretation of an experimental observation or a data set), there will be a much better chance that they will be able to do it on their own when the time comes.
When instructors first hear about active learning, many anticipate serious problems with it (I’ll never get through the syllabus if I do all that; the class will degenerate into chaos and I’ll never get control back; some students will refuse to participate; some will resent being asked to do anything…), and when they first use the method they may indeed encounter some student resistance and lack of participation. If you observe some precautions, however, and stay resolute for the first few weeks if you encounter resistance, those problems should become either nonexistent or inconsequential. For more ideas about what you might ask students to do in class and a rundown of what the precautions are, see Felder & Brent¹ and other papers you will find at <http://www.ncsu.edu/felder-public/Cooperative_Learning.html>

Collaborative/Cooperative Learning

**Collaborative learning** refers to two or more students working together on an assignment or project. There are several reasons for getting students to work collaboratively in lecture courses and not just in labs and the capstone design course, where collaboration is traditional. Engineering students will have to work in teams in their professional careers, and their performance evaluations could depend more on their ability to work well on those teams than on their technical skills. One of the mandated outcomes in the ABET Engineering Criteria is the ability to work in multidisciplinary teams, and students are unlikely to acquire high-level teamwork skills if they only work on teams in one or two courses. Perhaps most importantly, hundreds of research studies have shown that compared to students working individually, students working on well-functioning teams in a course learn more, learn at a deeper level, are less likely to drop out, and develop more positive attitudes toward the course subject and greater confidence in themselves.

Those benefits do not automatically occur whenever students work collaboratively, however, and most engineering graduates can tell horror stories about ineffective or dysfunctional teams. The most familiar problem involves “hitchhikers”—students who do little or nothing but get the same grade for the work as their more responsible teammates. Other common problems include dominant students who insist on doing everything themselves, students who are deliberately excluded for one reason or another, and interpersonal conflicts that arise because of different senses of responsibility, academic goals (high grades vs. passing grades), and personalities. When a team encounters those problems and cannot manage to resolve them, the members might well be better off working individually. Unfortunately, such situations frequently arise and quickly get out of hand when nothing is done to prevent them and to help students deal with them when they occur.

The way to maximize the benefits of teamwork is to use **cooperative learning**, a subset of collaborative learning in which the instructor builds in measures to assure that five conditions are met:

1. **Positive interdependence.** The students have to rely on one another for the effort to be successful.
2. **Individual accountability.** Each team member is held accountable for everything in the assignment or project, and not just the part for which he or she may have had primary

---

responsibility. If students hitchhike and don’t understand what the team did, they do not get credit for the work.

3. **Face-to-face interaction, at least part of the time.** Much of the learning in team projects takes place when the team meets to discuss, debate, and reach consensus on solutions to problems. If the team simply divides the work and staples the individual parts together without discussion, it is not cooperative learning.

4. **Facilitation of interpersonal skill development.** Students are not born with the project management, time management, communication, leadership, and conflict resolution skills needed to work effectively on a team. For team assignments to qualify as cooperative learning, the instructor must take steps to help the students develop those skills.

5. **Periodic self-assessment of team functioning.** At regular intervals, the teams must be required to reflect on what they are doing well as a team, what they need to work on to improve the team functioning, and what if anything they will do differently in the future.

Implementing cooperative learning effectively is not trivial. It requires knowing how to form teams and equip them to deal with the problems that commonly arise in teamwork, when to allow teams to dissolve and how to form new ones, how to structure assignments to assure both positive interdependence and individual accountability, and how to minimize or eliminate the resistance—and occasionally, the hostility—that some students feel toward instruction that requires them to work in teams. Suggestions regarding all of these points and links to the research base supporting cooperative learning may be found at <http://www.ncsu.edu/felder-public/Cooperative_Learning.html>. The monograph *Cooperative Learning in Technical Courses: Procedures, Pitfalls, and Payoffs* is a good place to start learning about the approach. Felder and Brent describe how the proper implementation of cooperative learning can equip students with all of the learning outcomes mandated by the ABET Engineering Criteria.

**Problem-Based Learning/Project-Based Learning and other Inductive Approaches**

The traditional approach to engineering instruction is deductive, proceeding from the general (principles and theories) to the specific (applications). In most courses, the instructor lectures on theories, principles, and mathematical methods and algorithms; gives assignments in which students practice the methods and algorithms; and later (sometimes much later) gets to applications. Engineering curricula work in much the same way. The students spend the first year learning basic science and math, then the next two learning mostly engineering science, and as seniors take the capstone design course in which they apply some of the fundamentals taught in the preceding three years to design a process or product.

The main problem with the deductive approach is that it is not how people normally acquire and retain new knowledge and skills. Rather, they do so by confronting problems that they need or want to solve; trying to accomplish their goal using what they already know and can do; discovering that more knowledge or skill is needed than they currently have and identifying what it is; gaining the required information (from books, classes, or observations of others solving similar problems) and adding it to their existing knowledge base; and practicing the required skills repeatedly and observing and reflecting on the outcomes of each attempt. In other words, people learn new material most effectively when they perceive a clear need to know it in order to solve a problem or meet a challenge. If they are simply presented with a body of new
material and told that in a month or in two or five years they’ll be shown why they need to know it, they are likely to learn it at best superficially.

An alternative and more effective instructional approach is to teach \textit{inductively}, presenting students with problems before they have been taught everything they need to know to solve them and then teaching the required material once the students can clearly see why they need to know it. There are many variations of this approach with different names and somewhat different emphases, including \textit{problem-based learning}, \textit{inquiry-based learning}, \textit{discovery learning}, \textit{need-to-know learning}, and \textit{just-in-time learning}. These methods are initially less comfortable for instructors than straightforward deductive presentation of material, and they can at first be distressing to students, who may not appreciate having to deal with problems they have not been taught to solve beforehand. Since induction is how people actually learn, however, the students taught this way are likely to end up with a much greater mastery of the knowledge and skills the instructor wishes to impart.

Formal problem-based learning calls for giving students significant problems whose solution requires the knowledge and skills normally taught in the course, and then having them work through the following steps, usually in teams:

1. Define the problem.
2. Build hypotheses to initiate the solution process.
3. Identify what is known, what must be determined, and what to do.
4. Generate possible solutions and decide on the best one.
5. Complete the best solution, test it, and either accept it or reject it and go back to Step 4.
6. Reflect on lessons learned.

The instructor serves primarily as a consultant, lecturing only when the need for new material arises in the context of the problem.

A related but less formal instructional approach is \textit{project-based learning}, which means that most of the learning in a course takes place in the context of projects, with lectures playing a subsidiary role or not taking place at all. The way the capstone design course is usually taught is project-based learning, as is the engineering laboratory in which each experiment can be considered a project. Several engineering departments have shifted some of their traditional lecture courses to project-based courses, and a few universities have made the switch for all of their courses, the best known of which is the University of Aalborg in Denmark. Whether project-based learning or one of the forms of problem-based learning is adopted, if student teams are involved, all of the methodologies of cooperative learning can be used to maximize the effectiveness of the approach.

Woods, Wankat, Duch, Groh, and Allen provide guidance on designing and implementing problem-based learning (Woods and Wankat are both engineering professors), and a collection of papers on engineering applications of the approach was recently published in the \textit{International Journal of Engineering Education} (vol. 19, #5, 2003). Felder and Brent describe how the proper implementation of PBL can equip students with all of the learning outcomes mandated by the ABET Engineering Criteria.
Learning (Instructional) Objectives

Learning objectives (aka instructional objectives) are statements of what students should be able to do if they have acquired the knowledge and skills the course is supposed to teach them. A learning objective takes one of the two following forms:

1. At the end of this [course, topic, chapter, lecture], the student should be able to…
2. To do well on the next test, you should be able to…

What follows either of these stems is a list of tasks that demonstrate mastery of the desired knowledge and skills. Each task statement includes one or more key action words [such as list, explain, calculate, estimate, derive, model, design, choose, and critique] along with a definition of the task and possibly a specification of the conditions under which the task is to be performed.

Following are examples of learning objectives that might appear on a study guide for an engineering test, with the key action words italicized.

To do well on the next test, you should be able to

1. Explain the statement, “The vapor pressure of pure water at 100°C is 760 mm Hg,” in terms that a bright high school student could understand.

2. Estimate the vapor pressure of a pure substance at a specified temperature or the boiling point at a specified pressure using (a) the Antoine equation, (b) the Cox chart, (c) the Clausius-Clapeyron equation and vapor pressures at two specified temperatures, (d) Table B.3 of your text. Rank-order your estimates in descending order of accuracy (best to worst), and briefly justify your ordering.

3. Given an equilibrium gas-liquid system with a single condensable component (A) and liquid A present, a correlation for the vapor pressure $p_A^*(T)$, and any two of the variables $y_A$ (mole fraction of A(v) in the gas phase), temperature, and total pressure, calculate the third variable using Raoult's law. List reasons why the calculated value might differ significantly from a measured value, assuming that the measurement is accurate.

4. For a process system that involves a gas phase containing a single condensable component and specified or requested values of feed or product stream saturation parameters (temperature, pressure, dew point, relative saturation or humidity, degrees of superheat, etc.), draw and label the flowchart, carry out the degree-of-freedom analysis, and perform the required calculations.

The action words in a learning objective must refer to observable actions—things an instructor could in principle watch the students doing. The words in the illustrative objectives just given meet this criterion, but words like learn, know, understand, and appreciate do not. You can’t watch someone understanding or appreciating something. If you want to know whether students understand a concept you have attempted to teach, you must ask them to do something observable that demonstrates their understanding. The things you might ask them to do would be your learning objectives for that concept.
All course instructors routinely write learning objectives, although most don’t call them that—they call them exams. Unfortunately, the first time many instructors seriously confront the question of what knowledge and skills they want their students to acquire is when they sit down to write the exams. That’s too late. The result is frequently that too much time is spent in lectures on material of secondary importance and too little is spent on things the instructor decides to emphasize on the tests—and students justifiably do not appreciate being taught one thing and tested on something else.

Having a good set of learning objectives in advance helps an instructor select course content and decide on how much time to allocate to each topic; plan lectures (talk about, illustrate, and give students active learning exercises in the things the instructor wants them to be able to do); create relevant assignments (give the students practice in those things); and write relevant tests (ask them to do some of the things). The objectives also do a much better job than the syllabus of defining the course to instructors preparing to teach it for the first time, instructors of prerequisite and subsequent courses in the curriculum, curriculum planning committees, and program accreditation visitors.

Learning objectives can be particularly valuable if they are shared with the students in the form of study guides for tests and then used as the basis of the test preparation. When students have a clear understanding of what is expected of them, there is a much greater chance that they will meet the expectations than if the expectations are muddy (as in, “Here is your 538-page text…you’re responsible for all of it…guess what I think is important enough to put on the test.”) Even if the study guides and tests include high-level thinking and problem-solving skills (as they should), the clarity of the expectations almost invariably leads to better student performance. A fringe benefit is that the instructor no longer has to deal with the ever-popular “Are we responsible for this on the test?” Once the students have the study guide, they know.

For more information on why and how to write learning objectives, see Felder and Brent,\textsuperscript{3,7} Gronlund,\textsuperscript{8} and Mager.\textsuperscript{9}

**Bloom’s Taxonomy**

When you start writing learning objectives, you quickly discover that different tasks call for dramatically different knowledge and skill levels, with some tasks requiring only rote memorization to complete and others calling for sophisticated analytical skills and creativity. A system of classifying learning objectives according to their required skill levels can help instructors make sure they are teaching and testing at an appropriate level for their students.

In the 1950s Benjamin Bloom and colleagues formulated such a system, called **Bloom’s Taxonomy of Educational Objectives**. Categories were formulated for cognitive (thinking and problem-solving skills), affective (attitudes, value systems), and psychomotor domains. The categories or *levels* for the cognitive domain and illustrative action words for each level are as follows:\textsuperscript{7,10}

1. **Knowledge** (repeating verbatim): *list* [the first ten alkanes]; *state* [the steps in the procedure for calibrating a gas chromatograph].
2. **Comprehension** (demonstrating understanding of terms and concepts): *explain* [in your own words the concept of vapor pressure]; *interpret* [the output from an ASPEN flowsheet simulation].

3. **Application** (applying learned information to solve a problem): *calculate* [the probability that two sample means will differ by more than 5%]; *solve* [the compressibility factor equation of state for \( P, T \), or \( V \) from given values of the other two].

4. **Analysis** (breaking things down into their elements, formulating theoretical explanations or mathematical or logical models for observed phenomena): *derive* [Poiseuille’s law for laminar Newtonian flow from a force balance]; *explain* [why we feel warm in 70°F air and cold in 70°F water].

5. **Synthesis** (creating something, combining elements in novel ways): *formulate* [a model-based alternative to the PID controller design presented in Wednesday’s lecture]; *make up* [a homework problem involving material we covered in class this week]; *design* [anything].

6. **Evaluation** (making and justifying value judgments or selections from among alternatives): *determine* [which of the given heat exchanger configurations is better, and explain your reasoning]; *select* [from among available options for expanding production capacity, and justify your choice]; *critique* [an essay, report, or article for accuracy and style].

Levels 4–6 are known as the **higher-level** (or **higher-order**) **thinking skills**.

All engineering instructors would say that they want their students to master higher-level thinking skills, but in many cases their lectures and homework assignments focus almost exclusively on Level 3. Then, if they put a high-level question on an exam (to see if the students “know how to think”) and the students do poorly on it, they blame it on the students’ lack of ability or poor study habits.

Their criticism is misdirected. The only way people acquire skills is through practice and feedback. If we teach at Level 3, it is unfair for us to require students to figure out for themselves how to work at Levels 4, 5, and 6, and especially unfair to expect them to figure it out on a calculation-packed 50-minute test. The best way to facilitate the development of higher-level skills is to include high-level tasks in learning objectives, share them with the students in study guides for exams, give illustrations and practice in class and more practice on assignments; and then put the high-level questions on the exams. If all that is done, most of the students who are capable of functioning at the high levels will be able to do so—and if engineering instructors collectively do it in every engineering course from the freshman through the senior year, our graduates will come out able to do modeling, design, and critical and creative thinking at a level that we can barely imagine now.

For information on writing learning objectives at all levels of Bloom’s Taxonomy, see Gronlund, Mager, and Besterfield-Sacre et al.

**Assessment and Evaluation.**

In engineering education it is frequently necessary to judge whether and how well students have learned a body of material or mastered a skill, or how well an instructor has taught a course, or
how well a product or process has met its design specifications, or how well an instructional program has met its educational objectives. A two-step process should be used to make the judgment rationally:

- **Assessment.** Decide on the data that will be used as a basis for making the judgment and the procedures (observations, measurements, experiments, surveys,…;) that will be used to obtain the data, then carry out the procedures and perform whatever analytical operations are needed to put the data into a form suitable for the next step.

- **Evaluation.** Using the assessment outcomes and pre-established criteria, draw inferences and make evaluative judgments. *(What grade does the student’s work deserve? Is the new laboratory course an improvement over the old one, and does the improvement justify the cost? Are the program graduates’ communication skills satisfactory? Should the paper be accepted for publication as is, or should it be rejected, or should it be sent back to the author for revision?)*

Assessment and evaluation have become extremely important in engineering education in the past decade—or to put it more accurately, their importance has become widely recognized. Program accreditation and the ABET Engineering Criteria are all about assessment and evaluation of learning. In addition, if you develop a new course or instructional software package or try an alternative teaching strategy in a class and you propose to submit a paper about it to the *Journal of Engineering Education*, the reviewers will immediately look for your assessment and evaluation plan. If it isn’t there or doesn’t stand up to their scrutiny, the paper will almost certainly be rejected no matter how clever your idea may be. *“We tried this method and we liked it and so did the students”* may have been acceptable ten or even five years ago, but it won’t cut it today. The same outcome will follow if you apply to the National Science Foundation for a CAREER Award or a grant to study new teaching materials or methods and you don’t have a solid assessment and evaluation plan built into the proposal. Funding agencies are not interested in financing ideas unless the principal investigator has a realistic plan to determine whether or not they work.

For a good introduction to assessment and evaluation of learning, see McKeachie, and for specific details on the assessment of engineering learning outcomes, see Felder and Brent and Besterfield-Sacre et al.

**ABET**

The *Accreditation Board for Engineering and Technology* (ABET) is the body that periodically reviews every engineering program (departments and interdisciplinary course programs) in the United States and determines whether they meet certain standards. Prior to a review of a program, the faculty assembles key information about the program’s educational goals, course offerings, faculty qualifications, and student products (homework, tests, reports, etc.) into a self-study: An ABET visitor (usually a faculty member from another institution) reviews the self-study, interviews the faculty and administrators, and decides whether the program should receive full (6-year) or probationary (3-year) accreditation or whether it should be denied accreditation.
The criteria ABET uses to make this determination (known as the Engineering Criteria) are both complex and flexible, and self-studies may vary considerably from one institution to another. The program is responsible for formulating its own goals or program educational objectives, making sure they reflect the mission of the university and the needs of the different program constituents (including students, faculty, and hirers of program graduates); program outcomes, or attributes of program graduates (knowledge, skills, values) that reflect the degree to which the program has met its objectives; outcome indicators, the assessment instruments and procedures that will be used to determine whether the graduates have achieved the outcomes; course learning objectives, statements of the things students should be able to do (define, explain, calculate, derive, model, design, evaluate,…) when they have completed each core course in the program curriculum; and a continuous improvement process that will be used to remedy any shortcomings revealed by the outcome assessments and continue to raise the program quality. The ABET visitor evaluates the appropriateness of the educational objectives, the extent to which the specified outcomes map onto the objectives and whether they incorporate eleven specific attributes specified by ABET (Outcomes 3a–3k), the extent to which the course learning objectives map onto the outcomes, the feasibility of the specified outcome assessment and continuous improvement processes, and the seriousness with which the program is implementing those processes.

The last paragraph only scratches the surface of the accreditation process and its jargon. For more details, see Felder and Brent.³

Last Words

The methods defined here only represent a start on all there is to know about effective teaching. Articles by the authors of this paper on many different aspects of engineering education can be found at <http://www.ncsu.edu/effective_teaching>, and two excellent general references on teaching and learning are McKeachie¹² and Wankat.⁵ If you pick up either of those books and randomly read a page, you are almost guaranteed to pick up a useful tip about some aspect of teaching along with information about the research that supports the tip.

The important thing to remember, though, is that learning how to teach as well as you’re capable of teaching is the work of a career—it’s not something to try to do in your first year. If you begin your next course determined to write a full set of instructional objectives before the first day of class and do a full-scale implementation of cooperative and problem-based learning, you will probably not be happy with the results. The time you need to devote to your research will be compromised, you’ll feel awkward and overstressed, and the students are likely to go into full-scale rebellion. Instead, pick one or two new methods (such as giving students study guides containing your instructional objectives and incorporating some active learning into your lectures), use them until you feel comfortable with them, and then gradually increase their use and add other new methods in subsequent courses, never venturing too far from your comfort zone. If you do that, your teaching will make a rapid initial improvement and will continue to improve thereafter, and that’s all you need.
References


RICHARD M. FELDER, Ph.D. (<www.ncsu.edu/effective_teaching>) is Hoechst Celanese Professor Emeritus of Chemical Engineering at North Carolina State University. He is co-author of *Elementary Principles of Chemical Processes* (Wiley, 2000), author or co-author of over 200 papers on engineering education and chemical process engineering, a Fellow Member of the ASEE, and co-director of the ASEE National Effective Teaching Institute.

REBECCA BRENT, Ed.D., is President of Education Designs, Inc., with interests that include faculty development in the sciences and engineering, support programs for new faculty members, preparation of alternative licensure teachers, and applications of technology in the K-12 classroom. She was formerly a professor of education at East Carolina University. She is co-director of the ASEE National Effective Teaching Institute.
When Linda Silverman and I wrote this paper in 1987, our goal was to offer some insights about teaching and learning based on Dr. Silverman’s expertise in educational psychology and my experience in engineering education that would be helpful to some of my fellow engineering professors. When the paper was published early in 1988, the response was astonishing. Almost immediately, reprint requests flooded in from all over the world. The paper started to be cited in the engineering education literature, then in the general science education literature; it was the first article cited in the premier issue of ERIC’s National Teaching and Learning Forum; and it was the most frequently cited paper in articles published in the Journal of Engineering Education over a 10-year period. A self-scoring web-based instrument called the Index of Learning Styles that assesses preferences on four scales of the learning style model developed in the paper currently gets about 100,000 hits a year and has been translated into half a dozen languages that I know about and probably more that I don’t, even though it has not yet been validated. The 1988 paper is still cited more than any other paper I have written, including more recent papers on learning styles.

A problem is that in recent years I have found reasons to make two significant changes in the model: dropping the inductive/deductive dimension, and changing the visual/auditory category to visual/verbal. (I will shortly explain both modifications.) When I set up my web site, I deliberately left the 1988 paper out of it, preferring that readers consult more recent articles on the subject that better reflected my current thinking. Since the paper seems to have acquired a life of its own, however, I decided to add it to the web site with this preface included to explain the changes. The paper is reproduced following the preface, unmodified from the original version except for changes in layout I made for reasons that would be known to anyone who has ever tried to scan a 3-column article with inserts and convert it into a Microsoft Word document.

Deletion of the inductive/deductive dimension

I have come to believe that while induction and deduction are indeed different learning preferences and different teaching approaches, the “best” method of teaching—at least below the graduate school level—is induction, whether it be called problem-based learning, discovery learning, inquiry learning, or some variation on those themes. On the other hand, the traditional college teaching method is deduction, starting with "fundamentals" and proceeding to applications.

The problem with inductive presentation is that it isn't concise and prescriptive—you have to take a thorny problem or a collection of observations or data and try to make sense of it. Many or most students would say that they prefer deductive presentation—“Just tell me exactly what I need to know for the test, not one word more or less.” (My speculation in the paper that more students would prefer induction was refuted by additional sampling.) I don't want

Author's Preface — June 2002
by Richard M. Felder
instructors to be able to determine somehow that their students prefer deductive presentation and use that result to justify continuing to use the traditional but less effective lecture paradigm in their courses and curricula. I have therefore omitted this dimension from the model.

**Change of the visual/auditory dimension to the visual/verbal dimension**

“Visual” information clearly includes pictures, diagrams, charts, plots, animations, etc., and “auditory” information clearly includes spoken words and other sounds. The one medium of information transmission that is not clear is written prose. It is perceived visually and so obviously cannot be categorized as auditory, but it is also a mistake to lump it into the visual category as though it were equivalent to a picture in transmitting information. Cognitive scientists have established that our brains generally convert written words into their spoken equivalents and process them in the same way that they process spoken words. Written words are therefore not equivalent to real visual information: to a visual learner, a picture is truly worth a thousand words, whether they are spoken or written. Making the learning style pair visual and verbal solves this problem by permitting spoken and written words to be included in the same category (verbal). For more details about the cognition studies that led to this conclusion, see R.M. Felder and E.R. Henriques, “Learning and Teaching Styles in Foreign and Second Language Education,” Foreign Language Annals, 28 (1), 21–31 (1995).


**The Index of Learning Styles**


And now, the paper.
“Professors confronted by low test grades, unresponsive or hostile classes, poor attendance and dropouts, know that something is wrong.” The authors explain what has happened and how to make it right.

Learning and Teaching Styles
In Engineering Education

Richard M. Felder, North Carolina State University
Linda K. Silverman, Institute for the Study of Advanced Development


Students learn in many ways—by seeing and hearing; reflecting and acting; reasoning logically and intuitively; memorizing and visualizing and drawing analogies and building mathematical models; steadily and in fits and starts. Teaching methods also vary. Some instructors lecture, others demonstrate or discuss; some focus on principles and others on applications; some emphasize memory and others understanding. How much a given student learns in a class is governed in part by that student’s native ability and prior preparation but also by the compatibility of his or her learning style and the instructor’s teaching style.

Mismatches exist between common learning styles of engineering students and traditional teaching styles of engineering professors. In consequence, students become bored and inattentive in class, do poorly on tests, get discouraged about the courses, the curriculum, and themselves, and in some cases change to other curricula or drop out of school. Professors, confronted by low test grades, unresponsive or hostile classes, poor attendance and dropouts, know something is not working; they may become overly critical of their students (making things even worse) or begin to wonder if they are in the right profession. Most seriously, society loses potentially excellent engineers.

In discussing this situation, we will explore:
1) Which aspects of learning style are particularly significant in engineering education?
2) Which learning styles are preferred by most students and which are favored by the teaching styles of most professors?
3) What can be done to reach students whose learning styles are not addressed by standard methods of engineering education?

Dimensions of Learning Style

Learning in a structured educational setting may be thought of as a two-step process involving the reception and processing of information. In the reception step, external information (observable through the senses) and internal information (arising introspectively) become available to students, who select the material they will process and ignore the rest. The processing step may involve simple memorization or inductive or deductive reasoning, reflection or action, and introspection or interaction with others. The outcome is that the material is either “learned” in one sense or another or not learned.

A learning-style model classifies students according to where they fit on a number of scales pertaining to the ways they receive and process information. A model intended to be particularly applicable to engineering education is proposed below. Also proposed is a parallel teaching-style model, which classifies instructional methods according to how well they address the proposed learning style components. The learning and teaching style dimensions that define the models are shown in the box.

Most of the learning and teaching style components parallel one another.* A student who favors intuitive over sensory perception, for example, would respond well to an instructor who emphasizes concepts (abstract content) rather than facts (concrete content); a student who favors visual perception would be most comfortable with an instructor who uses charts, pictures, and films.

* The one exception is the active/reflective learning style dimension and the active/passive teaching style dimension, which do not exactly correspond. The difference will later be explained.
The proposed learning style dimensions are neither original nor comprehensive. For example, the first dimension—sensing/intuition—is one of four dimensions of a well-known model based on Jung’s theory of psychological types, and the fourth dimension—active/reflective processing—is a component of a learning style model developed by Kolb. Other dimensions of these two models and dimensions of other models also play important roles in determining how a student receives and processes information. The hypothesis, however, is that engineering instructors who adapt their teaching style to include both poles of each of the given dimensions should come close to providing an optimal learning environment for most (if not all) students in a class.

There are 32 (2^5) learning styles in the proposed conceptual framework, (one, for example, is the sensory/auditory/deductive/active/sequential style). Most instructors would be intimidated by the prospect of trying to accommodate 32 diverse styles in a given class; fortunately, the task is not as formidable as it might at first appear. The usual methods of engineering education adequately address five categories (intuitive, auditory, deductive, reflective, and sequential), and effective teaching techniques substantially overlap the remaining categories. The addition of a relatively small number of teaching techniques to an instructor’s repertoire should therefore suffice to accommodate the learning styles of every student in the class. Defining these techniques is the principal objective of the remainder of this paper.

### Models of Learning & Teaching Styles

A student’s learning style may be defined in large part by the answers to five questions:

1) What type of information does the student preferentially perceive: **sensory** (external)—sights, sounds, physical sensations, or **intuitive** (internal)—possibilities, insights, hunches?

2) Through which sensory channel is external information most effectively perceived: **visual**—pictures, diagrams, graphs, demonstrations, or **auditory**—words, sounds? (Other sensory channels—touch, taste, and smell—are relatively unimportant in most educational environments and will not be considered here.)

3) With which organization of information is the student most comfortable: **inductive**—facts and observations are given, underlying principles are inferred, or **deductive**—principles are given, consequences and applications are deduced?

4) How does the student prefer to process information: **actively**—through engagement in physical activity or discussion, or **reflectively**—through introspection?

5) How does the student progress toward understanding: **sequentially**—in continual steps, or **globally**—in large jumps, holistically?

Teaching style may also be defined in terms of the answers to five questions:

1) What type of information is emphasized by the instructor: **concrete**—factual, or **abstract**—conceptual, theoretical?

2) What mode of presentation is stressed: **visual**—pictures, diagrams, films, demonstrations, or **verbal**—lectures, readings, discussions?

3) How is the presentation organized: **inductively**—phenomena leading to principles, or **deductively**—principles leading to phenomena?

4) What mode of student participation is facilitated by the presentation: **active**—students talk, move, reflect, or **passive**—students watch and listen?

5) What type of perspective is provided on the information presented: **sequential**—step-by-step progression (the trees), or **global**—context and relevance (the forest)?

### Dimensions of Learning and Teaching Styles

<table>
<thead>
<tr>
<th>Preferred Learning Style</th>
<th>Corresponding Teaching Style</th>
</tr>
</thead>
<tbody>
<tr>
<td>sensory perception</td>
<td>concrete content</td>
</tr>
<tr>
<td>intuitive</td>
<td></td>
</tr>
<tr>
<td>visual input</td>
<td>visual verbal presentation</td>
</tr>
<tr>
<td>auditory</td>
<td></td>
</tr>
<tr>
<td>inductive organization</td>
<td>inductive deductive organization</td>
</tr>
<tr>
<td>deductive</td>
<td></td>
</tr>
<tr>
<td>active processing</td>
<td>active student participation</td>
</tr>
<tr>
<td>reflective</td>
<td></td>
</tr>
<tr>
<td>sequential understanding</td>
<td>sequential perspective</td>
</tr>
<tr>
<td>global</td>
<td></td>
</tr>
</tbody>
</table>
Sensing and Intuitive Learners

In his theory of psychological types, Carl Jung introduced sensing and intuition as the two ways in which people tend to perceive the world. Sensing involves observing, gathering data through the senses; intuition involves indirect perception by way of the unconscious—speculation, imagination, hunches. Everyone uses both faculties, but most people tend to favor one over the other.

In the 1940s Isabel Briggs Myers developed the Myers-Briggs Type Indicator (MBTI), an instrument that measures, among other things, the degree to which an individual prefers sensing or intuition. In the succeeding decades the MBTI has been given to hundreds of thousands of people and the resulting profiles have been correlated with career preferences and aptitudes, management styles, learning styles, and various behavioral tendencies. The characteristics of intuitive and sensing types and the different ways in which sensors and intuitors approach learning have been studied.

Sensors like facts, data, and experimentation; intuitors prefer principles and theories. Sensors like solving problems by standard methods and dislike “surprises”; intuitors like innovation and dislike repetition. Sensors are patient with detail but do not like complications; intuitors are bored by detail and welcome complications. Sensors are good at memorizing facts; intuitors are good at grasping new concepts. Sensors are careful but may be slow; intuitors are quick but may be careless. These characteristics are tendencies of the two types, not invariable behavior patterns: any individual—even a strong sensor or intuitor—may manifest signs of either type on any given occasion.

An important distinction is that intuitors are more comfortable with symbols than are sensors. Since words are symbols, translating them into what they represent comes naturally to intuitors and is a struggle for sensors. Sensors’ slowness in translating words puts them at a disadvantage in timed tests: since they may have to read questions several times before beginning to answer them, they frequently run out of time. Intuitors may also do poorly on timed tests but for a different reason—their impatience with details may induce them to start answering questions before they have read them thoroughly and to make careless mistakes.

Most engineering courses other than laboratories emphasize concepts rather than facts and use primarily lectures and readings (words, symbols) to transmit information, and so favor intuitive learners. Several studies show that most professors are themselves intuitors. On the other hand, the majority of engineering students are sensors, suggesting a serious learning/teaching style mismatch in most engineering courses. The existence of the mismatch is substantiated by Godleski, who found that in both chemical and electrical engineering courses intuitive students almost invariably got higher grades than sensing students. The one exception was a senior course in chemical process design and cost estimation, which the author characterizes as a “solid sensing course” (i.e. one that involves facts and repetitive calculations by well-defined procedures as opposed to many new ideas and abstract concepts).

While sensors may not perform as well as intuitors in school, both types are capable of becoming fine engineers and are essential to engineering practice. Many engineering tasks require the awareness of surroundings, attentiveness to details, experimental thoroughness, and practicality that are the hallmarks of sensors; many other tasks require the creativity, theoretical ability, and talent at inspired guesswork that characterize intuitors.

To be effective, engineering education should reach both types, rather than directing itself primarily to intuitors. The material presented should be a blend of concrete information (facts, data, observable phenomena) and abstract concepts (principles, theories, mathematical models). The two teaching styles that correspond to the sensing and intuitive learning styles are therefore called concrete and abstract. Specific teaching methods that effectively address the educational needs of sensors and intuitors are listed in the summary.

Visual and Auditory Learners

The ways people receive information may be divided into three categories, sometimes referred to as modalities: visual—sights, pictures, diagrams, symbols; auditory—sounds, words; kinesthetic—taste, touch, and smell. An extensive body of research has established that most people learn most effectively with one of the three modalities and tend to miss or ignore information presented in either of the other two. There are thus visual, auditory, and kinesthetic learners. * Visual learners remember best what they see: pictures, diagrams, flow charts, time lines, films, demonstrations. If something is simply said to them they will probably forget it. Auditory learners remember much of what they hear and more of what they hear and then say. They get a lot out of discussion, prefer verbal explanation to visual demonstration, and learn effectively by explaining things to others.

Most people of college age and older are visual while most college teaching is verbal—the information presented is predominantly auditory (lecturing) or a visual representation of auditory information (words and mathematical symbols written in texts and handouts, on transparencies, or on a chalkboard). A second learning/teaching style mismatch thus exists.

* Concrete experience and abstract conceptualization are two poles of a learning style dimension in Kolb’s experiential learning model that are closely related to sensing and intuition.
this one between the preferred input modality of most students and the preferred presentation mode of most professors. Irrespective of the extent of the mismatch, presentations that use both visual and auditory modalities reinforce learning for all students.\textsuperscript{5,14,19,20} The point is made by a study carried out by the Socony-Vacuum Oil Company that concludes that students retain 10 percent of what they read, 26 percent of what they hear, 30 percent of what they see, 50 percent of what they see and hear, 70 percent of what they say, and 90 percent of what they say as they do something.\textsuperscript{21}

How to teach both visual and auditory learners: Few engineering instructors would have to modify what they usually do in order to present information auditorily: lectures accomplish this task. What must generally be added to accommodate all students is visual material—pictures, diagrams, sketches. Process flow charts, network diagrams, and logic or information flow charts should be used to illustrate complex processes or algorithms; mathematical functions should be illustrated by graphs; and films or live demonstrations of working processes should be presented whenever possible.

**Inductive and Deductive Learners**

*Induction* is a reasoning progression that proceeds from particulars (observations, measurements, data) to generalities (governing rules, laws, theories). *Deduction* proceeds in the opposite direction. In induction one infers principles; in deduction one deduces consequences.

*Induction* is the natural human learning style. Babies do not come into life with a set of general principles but rather observe the world around them and draw inferences: “If I throw my bottle and scream loudly, someone eventually shows up.” Most of what we learn on our own (as opposed to in class) originates in a real situation or problem that needs to be addressed and solved, not in a general principle; deduction may be part of the solution process but it is never the entire process.

On the other hand, *deduction is the natural human teaching style*, at least for technical subjects at the college level. Stating the governing principles and working down to the applications is an efficient and elegant way to organize and present material that is already understood. Consequently, most engineering curricula are laid out along deductive lines, beginning with “fundamentals” for sophomores and arriving at design and operations by the senior year. A similar progression is normally used to present material within individual courses: principles first, applications later (if ever).

Our informal surveys suggest that most engineering students view themselves as inductive learners. We also asked a group of engineering professors to identify their own learning and teaching styles: half of the 46 professors identified themselves as inductive and half as deductive learners, but all agreed that their teaching was almost purely deductive. To the extent that these results can be generalized, in the organization of information along inductive/deductive lines—as in the other dimensions discussed so far—a mismatch thus exists between the learning styles of most engineering students and the teaching style to which they are almost invariably exposed.
One problem with deductive presentation is that it gives a seriously misleading impression. When students see a perfectly ordered and concise exposition of a relatively complex derivation they tend to think that the author/instructor originally came up with the material in the same neat fashion, which they (the students) could never have done. They may then conclude that the course and perhaps the curriculum and the profession are beyond their abilities. They are correct in thinking that they could not have come up with that result in that manner; what they do not know is that neither could the professor nor the author the first time around. Unfortunately, students never get to see the real process—the false starts and blind alleys, the extensive trial-and-error efforts that eventually lead to the elegant presentation in the book or on the board. An element of inductive teaching is necessary for the instructor to be able to diminish the students’ awe and increase their realistic perceptions of problem-solving.

Much research supports the notion that the inductive teaching approach promotes effective learning. The benefits claimed for this approach include increased academic achievement and enhanced abstract reasoning skills; longer retention of information; improved ability to apply principles; confidence in problem-solving abilities; and increased capability for inventive thought.

Active and Reflective Learners

The complex mental processes by which perceived information is converted into knowledge can be conveniently grouped into two categories: active experimentation and reflective observation. Active experimentation involves doing something in the external world with the information—discussing it or explaining it or testing it in some way—and reflective observation involves examining and manipulating the information introspectively. *An “active learner” is someone who feels more comfortable with, or is better at, active experimentation than reflective observation, and conversely for a reflective learner. There are indications that engineers are more likely to be active than reflective learners.*

Active learners do not learn much in situations that require them to be passive, and reflective learners do not learn much in situations that provide no opportunity to think about the information being presented.

Active learners do not learn much in situations that require them to be passive, and reflective learners do not learn much in situations that provide no opportunity to think about the information being presented (such as most lectures). Active learners work well in groups; reflective learners work better by themselves or with at most one other person. Active learners tend to be experimentalists; reflective learners tend to be theoreticians.

At first glance there appears to be a considerable overlap between active learners and sensors, both of whom are involved in the external world of phenomena, and between reflective learners and intuitors, both of whom favor the internal world of abstraction. The categories are independent, however. The sensor preferentially selects information available in the external world but may process it either actively or reflectively, in the latter case by postulating explanations or interpretations, drawing analogies, or formulating models. Similarly, the intuitor selects information generated internally but may process it reflectively or actively, in the latter case by setting up an experiment to test out the idea or trying it out on a colleague.

In the list of teaching-style categories (table 1) the opposite of active is passive, not reflective, with both terms referring to the nature of student participation in class. “Active” signifies that students do something in class beyond simply listening and watching, e.g., discussing, questioning, arguing, brainstorming, or reflecting. Active student participation thus encompasses the learning processes of active experimentation and reflective observation. A class in which students are always passive is a class in which neither the active experimenter nor the reflective observer can learn effectively.
nately, most engineering classes fall into this category.

As is true of all the other learning-style dimensions, both active and reflective learners are needed as engineers. The reflective observers are the theoreticians, the mathematical modelers, the ones who can define the problems and propose possible solutions. The active experimenters are the ones who evaluate the ideas, design and carry out the experiments, and find the solutions that work—the organizers, the decision-makers. How to teach both active and reflective learners: Primarily, the instructor should alternate lectures with occasional pauses for thought (reflective) and brief discussion or problem-solving activities (active), and should present material that emphasizes both practical problem-solving (active) and fundamental understanding (reflective). An exceptionally effective technique for reaching active learners is to have students organize themselves at their desks in groups of three or four and periodically come up with collective answers to questions posed by the instructor. The groups may be given from 30 seconds to five minutes to do so, after which the answers are shared and discussed for as much or as little time as the instructor wishes to spend on the exercise. Besides forcing thought about the course material, such brainstorming exercises can indicate material that students don’t understand; provide a more congenial classroom environment than can be achieved with a formal lecture; and involve even the most introverted students, who would never participate in a full class discussion. One such exercise lasting no more than five minutes in the middle of a lecture period can make the entire period a stimulating and rewarding educational experience.31

Sequential and Global Learners

Most formal education involves the presentation of material in a logically ordered progression, with the pace of learning dictated by the clock and the calendar. When a body of material has been covered the students are tested on their mastery and then move to the next stage.

Some students are comfortable with this system; they learn sequentially, mastering the material more or less as it is presented. Others, however, cannot learn in this manner. They learn in fits and starts: they may be lost for days or weeks, unable to solve even the simplest problems or show even the most rudimentary understanding, until suddenly they “get it”—the light bulb flashes, the jigsaw puzzle comes together. They may then understand the material well enough to apply it to problems that leave most of the sequential learners baffled. These are the global learners.32

Sequential learners follow linear reasoning processes when solving problems; global learners make intuitive leaps and may be unable to explain how they came up with solutions. Sequential learners can work with material when they understand it partially or superficially, while global learners may have great difficulty doing so. Sequential learners may be strong in convergent thinking and analysis; global learners may be better at divergent thinking and synthesis. Sequential learners learn best when material is presented in a steady progression of complexity and difficulty; global learners sometimes do better by jumping directly to more complex and difficult material. School is often a difficult experience for global learners. Since they do not learn in a steady or predictable manner they tend to feel out-of-step with their fellow students and incapable of meeting the expectations of their teachers. They may feel stupid when they are struggling to master material with which most of their contemporaries seem to have little trouble. Some eventually become discouraged with education and drop out. However, global learners are the last students who should be lost to higher education and society. They are the synthesizers, the multidisciplinary researchers, the systems thinkers, the ones who see the connections no one else sees. They can be truly outstanding engineers—if they survive the educational process.

How to teach global learners: Everything required to meet the needs of sequential learners is already being done from first grade through graduate school: curricula are sequential, course syllabi are sequential, textbooks are sequential, and most teachers teach sequentially. To reach the global learners in a class, the instructor should provide the big picture or goal of a lesson before presenting the steps, doing as much as possible to establish the context and relevance of the subject matter and to relate it to the students’ experience. Applications and “what ifs” should be liberally furnished. The students should be given the freedom to devise their own methods of solving problems rather than being forced to adopt the professor’s strategy, and they should be exposed periodically to advanced concepts before these concepts would normally be introduced.

A particularly valuable way for instructors to serve the global learners in their classes, as well as the sequential learners, is to assign creativity exercises—problems that involve generating alternative solutions and bringing in material from other courses or disciplines—and to encourage students who show promise in solving them.3133 Another way to support global learners is to explain
Teaching Techniques to Address All Learning Styles

- Motivate learning. As much as possible, relate the material being presented to what has come before and what is still to come in the same course, to material in other courses, and particularly to the students’ personal experience (inductive/global).

- Provide a balance of concrete information (facts, data, real or hypothetical experiments and their results) (sensing) and abstract concepts (principles, theories, mathematical models) (intuitive).

- Balance material that emphasizes practical problem-solving methods (sensing/active) with material that emphasizes fundamental understanding (intuitive/reflective).

- Provide explicit illustrations of intuitive patterns (logical inference, pattern recognition, generalization) and sensing patterns (observation of surroundings, empirical experimentation, attention to detail), and encourage all students to exercise both patterns (sensing/intuitive). Do not expect either group to be able to exercise the other group’s processes immediately.

- Follow the scientific method in presenting theoretical material. Provide concrete examples of the phenomena the theory describes or predicts (sensing/deductive); then develop the theory or formulate the mod (intuitive/inductive/sequential); show how the theory or mod can be validated and deduce its consequences (deductive/sequential); and present applications (sensing/deductive/sequential).

- Use pictures, schematics, graphs, and simple sketches liberally before, during, and after the presentation of verbal material (sensing/visual). Show films (sensing/visual). Provide demonstrations (sensing/visual), hands-on, if possible (active).

- Use computer-assisted instruction—sensors respond very well to it (sensing/active).

- Do not fill every minute of class time lecturing and writing on the board. Provide intervals—however brief—for students to think about what they have been told (reflective).

- Provide opportunities for students to do something active besides transcribing notes. Small-group brainstorming activities that take no more than five minutes are extremely effective for this purpose (active).

- Assign some drill exercises to provide practice in the basic methods being taught (sensing/active/sequential) but do not overdo them (intuitive/reflective/global). Also provide some open-ended problems and exercises that call for analysis and synthesis (intuitive/reflective/global).

- Give students the option of cooperating on homework assignments to the greatest possible extent (active). Active learners generally learn best when they interact with others; if they are denied the opportunity to do so they are being deprived of their most effective learning tool.

- Applaud creative solutions, even incorrect ones (intuitive/global).

- Talk to students about learning styles, both in advising and in classes. Students are reassured to find their academic difficulties may not all be due to personal inadequacies. Explaining to struggling sensors or active or global learners how they learn most efficiently may be an important step in helping them reshape their learning experiences so that they can be successful (all types).

their learning process to them. While they are painfully aware of the drawbacks of their learning style, it is usually a revelation to them that they also enjoy advantages—that their creativity and breadth of vision can be exceptionally valuable to future employers and to society. If they can be helped to understand how their learning process works, they may become more comfortable with it, less critical of themselves for having it, and more positive about education in general. If they are given the opportunity to display their unique abilities and their efforts are encouraged in school, the chances of their developing and applying those abilities later in life will be substantially increased.

Conclusion

Learning styles of most engineering students and teaching styles of most engineering professors are incompatible in several dimensions. Many or most engineering students are visual, sensing, inductive, and active, and some of the most creative students are global; most engineering education is auditory, abstract (intuitive), deductive, passive, and sequential. These mismatches lead to poor student performance, professorial frustration, and a loss to society of many potentially excellent engineers.

Although the diverse styles with which students learn are numerous, the inclusion of a relatively small number of techniques in an instructor’s repertoire should be sufficient to meet the needs of most or all of the students in any class. The techniques and suggestions given on this page should serve this purpose.

Professors confronted with this list might feel that it is impossible to do all that in a course and still cover the syllabus. Their concern is not entirely unfounded: some of the recommended approaches—particularly those that involve the inductive organization of information and opportunities for student activity during class—may indeed add to the time it takes to present a given body of material.

The idea, however, is not to use all
A class in which students are always passive is a class in which neither the active experimenter nor the reflective observer can learn effectively. Unfortunately, most engineering classes fall into this category.

the techniques in every class but rather to pick several that look feasible and try them; keep the ones that work; drop the others; and try a few more in the next course. In this way a teaching style that is both effective for students and comfortable for the professor will evolve naturally and relatively painlessly, with a potentially dramatic effect on the quality of learning that subsequently occurs.

References

Teaching Architects and Engineers: Up and Down Bloom’s Taxonomy

Robert M. Arens, AIA
California Polytechnic State University at San Luis Obispo

Joseph P. Hanus, PE, PhD
United State Military Academy

Edmond Saliklis, PE, PhD
California Polytechnic State University at San Luis Obispo

Abstract

Engineering faculty and Architecture faculty both address student learning through the prism of Bloom’s taxonomy of the cognitive domain, but do so in diametrically opposite manners. Engineering faculty tend to assess student learning starting at the lowest taxonomy level, Acquisition of Knowledge, and progress in their curriculum and courses to the higher levels of Synthesis and Evaluation. Compare this to a studio environment in an undergraduate Architecture curriculum, where the faculty often begin with the highest levels, such as Evaluation in applying value judgments about the adequacy of the design and Synthesis, by putting disparate pieces of information together, and Analysis in solving large complex problems by reducing them to smaller pieces. Thus, the paper’s hypothesis is that Engineering faculty typically move up Bloom’s taxonomy of the cognitive domain, whereas Architecture faculty typically move down the taxonomy.

The implications of this hypothesis are interesting from both a pedagogical and practical point of view. Can we learn from each other and benefit from each other’s experience? Can we aid the students who seek larger global understanding, yet are often discouraged during their preliminary acquisition of fundamental factual knowledge?

This paper explores this thesis by studying the literature surrounding the Cognitive Domain in both Civil Engineering and Architecture, and gives some suggestions for providing engineering students with more opportunities to explore higher levels on Bloom’s taxonomy in the undergraduate curriculum.

Introduction

The authors have acted as guest jurors in each other’s courses when students have made public presentations of the work, otherwise known as the critique (or final crit). A striking revelation was made to the engineer that in an architecture critique, many of the issues brought up by jurors and by the student peers appeared to touch on relatively high level concepts in Bloom’s Taxonomy of Learning. The taxonomies are a language that is proposed to describe the progressive development of an individual’s cognitive understanding of material.

Thus, this paper began as an exploration of the thesis that Architecture faculty are comfortable moving up and down the continuum of Bloom’s Taxonomy, whereas Civil Engineering faculty traditionally move up from the lowest levels of the taxonomy and they are challenged to reach the higher levels with their students.

The purpose of this paper is to review the literature that might support this thesis, and to recommend how Civil Engineering faculty might learn to move up and down the taxonomy from their Architectural peers.

Bloom’s Taxonomy

Bloom’s Taxonomy is the seminal work of the 1950’s educational committee chaired by Benjamin Bloom. The committee established a set of taxonomies in three domains of learning: cognitive, affective and psychomotor.
The cognitive domain taxonomy is widely accepted in many fields and has been identified as, “arguably one of the most influential education monographs of the past half century.”\(^1\) The taxonomies are a language that is proposed to describe the progressive development of an individual in each domain and are defined as follows:

Cognitive: of, relating to, being, or involving conscious intellectual activity.

Affective: relating to, arising from, or influencing feelings or emotions.

Psychomotor: of or relating to motor action directly proceeding from mental activity.\(^2\)

The domains and their development levels (based on work by Bloom\(^3\), Krathwohl et al.\(^4\), and Simpson\(^5\)) are shown in Figure 1. In this paper, we will focus exclusively on the Cognitive Domain, and we will refer to the ranking of these as Levels 1 (Knowledge) through Level 6 (Evaluation).

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
\textbf{Cognitive Domain}\(^1\) & \textbf{Affective Domain}\(^2\) & \textbf{Psychomotor Domain}\(^3\) \\
Knowledge & Receiving & Perception \\
Comprehension & Responding & Set \\
Application & Valuing & Guided Response \\
Analysis & Organization & Mechanism \\
Synthesis & Characterization by a Value & Complex Overt Response \\
Evaluation & Complex & Adaptation \\
 & & Origination \\
\hline
\end{tabular}
\caption{The Domains and their levels from Bloom’s Taxonomy.}
\end{table}

The authors propose that traditional undergraduate programs in Civil Engineering begin at the lower level of Bloom’s and work up, yet may not achieve the highest levels. This position is based upon review of the American Society of Civil Engineers’ (ASCE) Civil Engineering Body of Knowledge for the 21st Century, a review of the literature describing such programs, and the authors’ experience in our own schools.

The Civil Engineering “Body of Knowledge (BOK) for the 21st Century”\(^6\) was prepared by BOK Committee of the Committee on Academic Prerequisites for Professional Practice. In this report, fifteen Accreditation Board for Engineering and Technology (ABET) outcomes were linked to the six levels of Bloom’s taxonomy for cognitive development, as shown in Figure 1. It is clear from this report that undergraduate Civil Engineering programs are primarily focused on the lower levels of achievement in terms of Bloom’s taxonomy. It is noteworthy that the level of achievement expected for the “Design” outcome does not include Evaluation (see line 9 in Figure 1). In Figure 1, the icon “E” stands for fulfillment via post-baccalaureate, pre-licensure work experience.

Engineering programs commonly employ a capstone course as the culminating event for their students’ development – particularly in design. The 2005 national survey of capstone design courses by Howe and Wibarger\(^7\) reported on 444 programs from 232 institutions. This survey found that 79% of institutions reported a one, or two course sequence for their capstone experience. The premise of such end-of-program culminating experiences was that students eventually gained proficiency at the “design” level, apparently reaching higher level of Bloom’s. We note however, that recently there is a trend for curricula to move towards integrating the design experience earlier in the students’ program; thereby reaching higher cognitive levels in courses other than the capstone experience. Examples include the “design focus curriculum” at Olin\(^8\), a pre-capstone approach at Oklahoma State University\(^9\), and the emerging inclusion of freshman engineering courses such as those at the University of Southern Indiana\(^10\). However, it is the still capstone experience that many programs use\(^11\) to assess their ABET Design Outcome\(^12\).

PART 1:
The Civil Engineering Undergraduate Curriculum

The authors’ own experiences contributed to the realization of the difference between Civil Engineering and Architecture programs, with respect to Bloom’s taxonomy. Civil Engineering programs are typically formulated as one might design an actual building, that is, they are built from the ground up. CE programs begin with the design and analysis of individual components in Statics, Mechanics of Materials, etc. Those components are then combined to form sub-systems and eventually are fully integrated with reference to design codes. It is not till late in most CE programs that students grapple with the complete design of buildings and structures. Clearly this stems from that fact that most CE educators believe...
one must “build civil engineers” from the ground-up, as one would build a building. However, the authors have come to appreciate that a different model, and potentially a more fruitful and pedagogically sound model, can be created by emulating the best practices from Architecture programs.

**PART 2:**

**The Undergraduate Architecture Studio**

In his influential book *Educating the Reflective Practitioner* (1987), Donald Schön, argues that professional education should be centered less on developing a specific set of skills in students and more on their ability to reflect first, then act in situations where established theories may not apply. He addresses the implications of the “ground-up” approach to educate Civil Engineers mentioned above, when he writes, “Civil engineers know how to build roads suited to particular sites and specifications. They draw on their knowledge of soil conditions, materials, and construction technologies to define grades, surfaces, and dimensions. When they must decide what road to build, however, or whether to build it at all, their problem is not solvable by the application of technical knowledge, not even by the sophisticated techniques of decision theory. They face a complex and ill-defined mélange of topographical, financial, economic, environmental, and political factors. If they are to get a well-formed problem matched to their familiar theories and techniques, they must construct it from the materials of the situation…”

The ground-up approach, it seems, would prepare students if they only work with straightforward, well-formed cases and problems to which they can apply standard theories. “But as we have come to see with increasing clarity over the last twenty or so years,”
Schön continues, “the problems of real-world practice do not present themselves to practitioners as well-formed structures. Indeed, they tend not to present themselves as problems at all but as messy, indeterminate situations.” This ability to “construct the problem” is precisely the type of skill addressed by the higher levels of Bloom’s taxonomy. In other words, the very ability that aspiring engineers need most may be the skill that professional schools seem less able to teach.

Of course, the “ground-up” approach is not unique to CE programs. It characterizes many professional schools in the University setting. Schön believes that most professional programs are premised on technical rationality, due in part to a desire to gain prestige from the science/research communities when joining universities at the beginning of the twentieth century. He writes, “their normative curriculum…still embodies the idea that practical competence becomes professional when its instrumental problem-solving is grounded in systematic, preferably scientific knowledge. So the normative curriculum presents first the relevant basic science, then the relevant applied science, and finally, a practicum in which students are presumed to learn to apply research-based knowledge to the problems of everyday practice.” In other words, the curricula of most professional programs are premised on taking students to the level of Application (Level 3) in Bloom’s taxonomy but no higher.

What does Schön suggest as a course of action for professional programs? He suggests that practitioners would be far more competent in indeterminate zones of practice if they became more like artists (he defines artistry as “an exercise of intelligence, a kind of knowing, though different in crucial respects from our standard model of professional knowledge. It is not inherently mysterious; it is rigorous in its own terms…”) We will return to this key of idea of rigor in the third and final part of our paper. By looking at the skills of extraordinarily gifted practitioners and by assessing how these masters acquired such skills, Schön realized that professional artistry was best fostered under conditions similar to those in art studios and music conservatories, namely environments where students “learn by doing” in a relatively low risk situation, where just about everything is practicum, and where they have access to mentors who coach more
than teach. After having the chance to observe architectural education firsthand, he became convinced that “architectural designing is a prototype of the kind of artistry that other professionals need most to acquire; and the design studio, with its characteristic pattern of learning by doing and coaching, exemplifies the predicaments inherent in any reflective practicum and the conditions and process essential its success. Thus, other professional schools can learn from architecture”17.

Whereas Schön believes the design studio may be a model for all of the professions, we argue that it is particularly well-suited for the education of engineers due to its attempt to blend both art and science in the “learn-by-doing” experience. Of architecture programs and the education they provide, Schön writes, “they are interesting because they occupy a middle ground between professional and art schools. Architecture is an established profession charged with important social functions, but it is also a fine art, and the arts tend to sit uneasily in the contemporary research university. In their curricula, some applied sciences may be taught, although the status of such sciences is often ambiguous and controversial. For the most part, however, these schools preserve a studio tradition centered on the art of designing”18.

How do architecture programs use the studio model? Generally most schools offering an undergraduate degree in architecture introduce students to building design studios in the second year of a five-year program. These studios typically present students with a hypothetical building project (e.g. design an art gallery with living spaces for the gallery owner for a vacant infill site in San Francisco’s SOMA district) and are guided through its design by the studio instructor. At the end of the process (and sometimes at several intermediate points) outside critics are invited in to offer criticism, insights and advice to help the students in their progress. This model is repeated throughout the students’ education, with the complexity of the project components (site, building type, construction systems, etc.) increasing as students progress through the program.

It is typical for Architecture students to initially be guided by both instructors and critics via an analysis or “reading” of the project, i.e. a thorough understanding of the project’s social, environmental and programmatic context. This equates to the formation of a general understanding of project determinants, influences and parameters before the proposal of any specific design. The pedagogy that supports this is that design should be informed by the project’s broad context, and that a design proposal should not move into specific terms until general terms are vetted. Upon completion of this analysis phase, students are encouraged to formulate a conceptual framework for their project that sometimes draws from wide sources of inspiration or influence, often beyond the discipline of architecture. These sources may include disciplines such as philosophy, literature, biology, and others. How refreshingly different this is from the traditional Civil Engineering model!

In the paper “Models of Design in Studio Teaching” (1985), Stefani Ledewitz refers to the process described above as the “Analysis/Synthesis Model” and argues that it is the major component of the design studio experience. Ledewitz writes of this model, “A studio project is often divided into two discrete and identifiable parts. The first part, which might take from a few days to many weeks, is the analysis phase, in which site, program, building type, context and other investigations are carried out...At some point, the studio shifts in focus to the design concept, and assignments change from analytic exercises to design proposals. During this stage, references are made back to analysis work.”19
Ledewitz goes on to argue that “all the aspects of design education—the skills, the language, and the approach to problems—are more effectively taught indirectly through experience than taught directly by explanation”. This is not the engineering approach! The ability to “think architecturally” is the most difficult to explain to a student who lacks design experience, yet this is a primary goal of the undergraduate architectural design studio.

The very words “Analysis” (Level 4) and “Synthesis” (Level 5) are telling in this study. Ledewitz describes the typical design studio as moving up and down taxonomy to touch these two levels. Ledewitz also states that at the introductory studio stage it is unlikely that design instructors would impose upon a student a predefined procedure to solve a problem. Again, this is totally different from the engineering approach wherein textbooks often include a map of strategies for how to solve various problems.

A strong endorsement for the studio pedagogy is made by Ernest L. Boyer and Lee D. Mitgang in their study of architectural education sponsored by the Carnegie Foundation for the Advancement of Teaching (1996). In it they wrote that “architecture education, at its best, is a model that holds valuable insights and lessons for all of higher education as a new century approaches…in short, architecture education is really about fostering the learning habits needed for the discovery, integration, application and sharing of knowledge over a lifetime.”

PART 3: Synthesis and Suggestions

We close by offering a suggestion of how to integrate the best practices of the undergraduate architecture design studio with the traditional undergraduate Civil Engineering curriculum. We begin by analyzing the word “design”.

In his book The Aesthetics of Architecture, Roger Scruton argues that architecture concerns itself with the expression of a set of “values”. He writes, “A value, unlike a mere preference, expresses itself in language…and it pursues what is right, fitting, appropriate and just. A value is characterized not by its strength but by its depth, by the extent to which it brings order to experience… Values are a special case of ends of conduct; they define what we are aiming at, not just in the particular case, but generally. It is through the acquisitions of values that we are able to arrive at a conception of an end… To have such a “conception of an end” enables one to envisage what it would be like to achieve that end.”

When Scruton uses the phrase “conception of an end” or to “envision what it would be like”, he is arguing that the acquisition of “design values” is partly imaginative, requiring envisaging a non-existing state of affairs. It is also evaluative, i.e. it involves a sense of appropriateness of one’s actions. What a fascinating definition of design this is! If we could only encourage our students to develop such a rigorous design ethic, one that seeks to impart order on an as-yet unbuilt project, it would nurture a future generation of leading thinkers in structural design. Clearly such a design ethic, or set of values, requires high levels of cognition on Bloom’s taxonomy.

Jones (1981) has analyzed Scruton’s quote regarding the attachment of value to a series of “ends”. Jones has argued that architecture students must be educated in the appreciation of a vast array of accomplished “ends”, as well as in “the imaginative construction of ends yet to be”. Jones goes on to argue that the undergraduate study of architecture must “establish a balance between the appreciation of the socio-cultural process of expressing ‘ends in view’ and the techniques of building these.”

Surely, this line of thinking is reflected in the two accrediting agencies, ABET for the structural engineers and NAAB for the architects. ABET endorses a linear, progressive march through higher and higher taxonomy levels, essentially advocating the laying of foundations for “techniques” of the “ends in view”. NAAB (2006) advocates tipping the balance in favor of establishing an “appreciation of the socio-cultural processes” surrounding these “ends in view”. Another interesting insight occurs when we consider that ABET relegates experience of the highest taxonomy to the post-B.S. workplace environment, whereas NAAB actively promotes the idea that the architecture student will master the lowest taxonomy levels through his or her Intern Development Program (IDP), post B.Arch. The IDP carefully monitors competence in basic comprehension of a wide range of practical architectural experiences prior to allowing the junior architect to sit for the licensing exam.

Conclusions

We close by advocating that in upper level interdisciplinary or in capstone projects, that faculty encourage students to explore both ends of the taxonomy. A practical way of ensuring this is to continually nurture in the students a sense of appreciation of the context of their work, both historical and contemporary/global, along with guidance in the techniques necessary to achieve
these ends. A pedagogically sound way of achieving this goal is to implement the Analysis/Synthesis Model we have described, and to encourage undergraduate students to develop and to articulate a set of “design values”. Our findings have convinced us that it is through our careful mentoring of our students’ public articulation of “design values”, such as those in a studio critique, that we will encourage them to develop a sense of design ethics. Brilliant structural designers of the past century such as Robert Maillart, Felix Candela, Pier Luigi Nervi and Fazlur Khan have all written extensively about the “structural logic” that has informed their design worldview. These giants of the past still retain the power to inspire and to challenge the structural designers of the future. Each of these exemplary designers had a strong and clear set of design values which framed their ground breaking designs. It is our challenge as educators to present these historically significant ideas in a new 21st model of engineering education that incorporates the best features of the undergraduate Architecture design studio.

Notes
15. Ibid. p. 8.
17. Ibid. p. 18.
18. Ibid. p. 18.
20. Ibid. p. 2.
Using Classical Mechanism Concepts to Motivate Modern Mechanism Analysis and Synthesis Methods

Robert LeMaster, Ph.D.¹

Abstract

This paper describes a methodology by which fundamental concepts in the study of mechanisms are used to introduce students to a broad range of topics ranging from the classical graphical methods to state-of-the-art multi-body software. In this method, two central concepts associated with multi-body simulation software are used as unifying threads. The central concepts are: 1) the role of constraint equations, and 2) the difference between inverse-dynamic and dynamic analyses. These central concepts allow a variety of methods to be built on each other in a progressive and unified approach. Students taking a course based on this method are exposed to state-of-the-art technology without sacrificing traditional course content.

Introduction

Most undergraduate curricula in mechanical engineering include a one-semester course in kinematics and dynamics of machinery. The intent of this first course is to teach students how concepts learned in a prerequisite dynamics course can be applied to the analysis and design of mechanisms or machine components. Traditionally, this first course has included an introduction to the kinematic and kinetic analysis of mechanisms using graphical methods.

In recent years, mechanism simulation software has also been used in these courses. The simulation software can be classified into two different types. The first type is limited to a particular type of mechanism (e.g. four-bar, slider-crank, etc), and is usually distributed with textbooks [1]. This type of software can usually animate the motion of the mechanism, and enables students to quickly observe the effects of changing various parameters. These programs are very easy to use and have good user interfaces. The major drawback with programs dedicated to a particular type of mechanism is that they do not expose students to the modeling concepts used by the general-purpose software found in industry.

The second type of software is general-purpose multi-body simulation software. This powerful software enables engineers to simulate complex mechanical systems accurately and efficiently. The graphical user interfaces integrated with these programs enable complex geometries to be developed quickly, and the animation of simulation results provides tremendous insight into the interaction of the different components. The multi-body simulation programs commonly used in the United States includes ADAMS, DADS, and

¹ Department of Engineering, College of Engineering and Natural Sciences, University of Tennessee at Martin, Martin, TN.
Working Model. In addition to these programs, popular CAD programs such as CATIA, I-DEAS, ProEngineer, and Unigraphics, have multi-body modeling and simulation options.

Multi-body simulation software represents the state-of-the-art in the analysis and synthesis of mechanical systems. The theoretical development of the equations used in these programs involves the development and solution of a complex set of differential-algebraic equations. The development of these equations and the study of methods for solving them are typically not covered at the undergraduate level. In fact, courses that focus on the development of the equations, their computer implementation, and solution are taught in graduate level courses, if at all.

Since employers expect students to have familiarity with the tools and methods used in industry, it is important to expose students to state-of-the-art tools and methods. There is a danger in this exposure, in that learning to run a specific simulation program is a skill that can be accomplished with limited understanding of the underlying mathematics or methods used by the program. Therefore, if too much emphasis has been placed on learning how to run a simulation program, the students will have acquired a skill, but will have gained little knowledge. The challenge faced by instructors teaching a first course in the kinematics and dynamics of machines is how to best introduce students to the state-of-the art technology, while at the same time ensuring that they have a grasp on the fundamental concepts, methods, and equations traditionally taught in undergraduate mechanism analysis and design courses.

The purpose of this paper is to present an instructional approach that uses classical mechanism concepts to motivate and provide insight into the workings of state-of-the-art programs. It identifies concepts that are central to both classical and computer based methods, and describes how emphasis is placed on a few central concepts as students progressively move through classical graphical methods, MATLAB simulations, and multi-body simulations using I-DEAS.

There are two central concepts that the author emphasizes to students taking a first course in kinematics/dynamics of machines. These central concepts provide a common theme that tie seemingly different methods or topics together as the course progresses. The motivation for these central concepts is presented in the following paragraphs.

**Central Concepts**

The algebraic-differential equations that control the motion of a mechanism consist of two parts: 1) the nonlinear algebraic constraint equations, and 2) the differential equations of motion. These algebraic and differential equations are coupled and various solution methods can be employed in their solution [2]. Two distinct types of problems can be encountered during the study of mechanisms: 1) problems requiring the direct solution of the algebraic-differential equations, and 2) problems requiring the solution of the kinematic constraint equations prior to computing the forces necessary to cause the motion. Huang [2] refers to the first type of problem as a dynamic problem, while the second type is called an inverse-dynamic problem. Multi-body simulation programs generally have the ability to solve either type of problem. Therefore, the ability to recognize the type of problem being solved is a “central concept”.

Whether a dynamic or inverse-dynamic problem is being solved, constraint equations must be enforced as the mechanism moves. Therefore, constraint equations represent the second “central concept”. As we will see, traditional analysis methods that students are introduced to in a first course in kinematics and dynamics of machines use constraint equations quite
extensively. However, textbooks rarely mention this unifying or central concept as the various methods are presented, and students see merely a hodge-podge of unrelated topics. In fact, a review of several kinematics textbooks revealed that the word constraint was not used or was used only when discussing the concept of degrees-of-freedom of a mechanism. [1,3-4]

**Constraint Equations – A Unifying Thread**

The most common feature of all mechanism analysis and design methods is the concept of constraint equations. In their simplest form, constraint equations are algebraic equations that define the position and orientation of all bodies in a mechanism as the mechanism moves. The concept of constraint equations is so fundamental that it can be used as a unifying thread to introduce students to a broad spectrum of topics ranging from graphical methods to multi-body simulation software.

Two types of constraints must be introduced when exposing students to constraint equations. The first type of constraint is a “drive” constraint. This type of constraint gives the position or orientation of a body in the mechanism as a function of time. In the case of a four-bar mechanism, the drive constraint specifies the angular orientation of the drive link as a function of time. The second type of constraint is a “kinematic” constraint. Kinematic constraint equations are mathematical relationships that relate the position and orientation of bodies in a mechanism. The solution of any mechanism problem by any method requires the development of these relationships.

Students are first introduced to constraint equations through the development of “circle-based” constraint equations for a four-bar linkage, Fig. 1. These equations are the simplest form of the equations governing four-bar mechanisms and are the easiest for students to understand. The next step is to introduce students to two important methods for solving these equations. The first solution method is to solve the equations graphically. AutoCAD is used as a tool while performing a graphical solution. As part of the graphical solution method, the Grashof criteria are introduced and are used to demonstrate that there are situations under which solutions of the type desired do not exist.

The graphical position analysis of mechanisms has been included in kinematics courses for many years. The author believes that graphical methods (particularly position based methods) continue to play an important role in the design of mechanisms. The author has observed that in practice, an engineer will often start the design of a mechanism using graphical position analysis methods. These methods often provide the most direct route to identifying a possible design. Once an initial design concept has been achieved, the engineer may move on to more sophisticated analysis and design tools depending on the problem at hand.

The second solution method introduced as a means for solving the “circle-based” equations is the Newton-Raphson method. Students are introduced to the method and are shown how to find the Jacobian matrix. A Matlab program written by the author is examined in detail and used as a demonstration of how to implement the method numerically. The importance of the determinant of the Jacobian matrix in determining whether or not a solution exists is discussed and is related to the Grashof criteria that was developed using graphical methods.

At this point, students have been introduced to the concept of constraint equations and have seen two methods for solving them – graphical and numerical. The concept of solution feasibility has been introduced and demonstrated for both methods. Vector loop equations are next introduced as a method for developing kinematic constraint equations. Instead of
locating points at the intersection of circles, vector equations are used to develop the kinematic constraint equations. Position vector loop equations are developed for the four-bar linkage and students are given an assignment that requires them to develop a MATLAB program that solves the vector loop equations using the Newton-Raphson method. This program is very similar to the one developed by the instructor to solve the “circle-based” equations, but involves a different set of equations and associated Jacobian matrix. The results of this exercise are compared to those obtained using the “circle” equations – both numerical and graphical.

![Figure 1. Constraint Circles for Four-bar Mechanism. The coupler and follower joint are always located at the intersection of the circles. The center of rotation of the coupler circle moves with the drive link.](image)

The development of constraint equations for mechanisms other than the four-bar mechanism is next introduced. In-class examples are worked showing how to write and solve constraint equations for mechanisms such as the slider-crank, Geneva wheel, Whitworth Quick-Return, and Swing-Arm Quick-Return. Students are given homework assignments that require them to test their understanding and to develop their skill. In the assignments, students must develop the constraint equations and solve them numerically. They are then required to verify their answers using a graphical position analysis. This dual method approach reinforces the relationship between graphical and numerical solution methods and assists students in trouble-shooting their answers. It also reinforces the importance of anchoring or verifying numerical solutions.

Students initially have difficulty writing the constraint equations, and it is important to follow up these assignments by going over the solutions in class. The author has observed that students struggle with writing the constraint equations because they have trouble visualizing the geometrical relationships involved. The concepts introduced in graphical position analysis are useful in helping students “see” these relationships. As a first step in writing constraint equations, students are instructed to visualize how they would perform a graphical position analysis. Generally, this involves finding the intersection points of lines.
and/or circles or vector relationships. Next, they are instructed to write down the equations that describe the geometric entities used in performing the graphical position analysis. Once they “see” the geometric constraints that must be satisfied (graphical position analysis), it is easier to write down and solve the equations that describe constraints (numerical position analysis).

**Velocity and Acceleration Analysis**

The central role of constraint equations is further demonstrated by showing how they are used to find the velocity and acceleration. The concepts of holonomic constraints and generalized coordinates are introduced. These concepts are easily understood at this point because they can be related to what the students have previously seen. In the case of the four-bar mechanism, the generalized coordinates are the angles associated with the vector loop equations or the Cartesian coordinates of the joints in the case of the circle-equations.

The time rate-of-change of the generalized coordinates are found by taking the time derivative of the constraint equations using the chain rule of differentiation. The angular velocities of links and linear velocity of joints are then related to the time rate of change of the generalized coordinates. In a similar manner, the accelerations are found by taking the time rate of change of the velocity equations [2]. Students are required to add the velocity and acceleration equations to the MATLAB program developed during previous assignments, and are required to verify their answers using graphical methods that are also developed during this portion of the course.

**Dynamic and Inverse-Dynamic Analysis**

The equations of motion for a single link in the mechanism are developed and methods for assembling the equations for a complete mechanism are presented in class. As in similar situations, a MATLAB program developed by the instructor is reviewed and students are required to add the ability to compute joint forces to their programs. These forces are computed using a drive constraint (inverse-dynamic analysis). The difference between a dynamic analysis and inverse-dynamic analysis is experienced by having the students attempt to find the angular velocity of the drive link for a situation in which the torque acting on the drive link is specified instead of the angular position versus time. Once students have determined that they don’t know how or that it can’t be done solving the algebraic constraint equations, the difference between inverse-dynamic analysis and dynamic analysis is explained. Although not examined in detail, Adam-Moulton based methods for solving this type of problems are briefly discussed.

Next, the method for designing a flywheel using a torque-versus-angular position curve is developed. Students use their MATLAB four-bar programs to compute the torque-position curve and design a flywheel. They are then asked to add the flywheel mass and inertia properties to their programs to see the change in torque variation. Since their programs are based on a constant angular velocity constraint equation, they again run into a practical problem that requires a full dynamic analysis (i.e. find the angular velocity of the drive link resulting from an input torque). This process of having students encounter a problem and attempt to explain why what they have learned falls short has proven to be a very good approach for teaching the difference between inverse-dynamic and dynamic analyses. Recall that the ability to differentiate between these two types of analyses was one of the central concepts that the author believes is necessary to understanding the features of multi-body simulation programs.
Multi-body Simulation Software Application

At this point in a course, students have a good understanding of the importance of constraint equations to mechanism analysis and design. They have written constraint equations for a variety of mechanisms and have solved the equations both graphically and numerically. They have seen how the constraint equations are used to find velocities, accelerations, and forces. They also understand the difference between an inverse-dynamic analysis and a dynamic analysis. Students are now introduced to kinematic simulation using the mechanism option in the I-DEAS software marketed by SDRC, Inc.

The instructor demonstrates the use of the software by going through the steps necessary to model and analyze the four-bar mechanism shown in Fig. 2. During this demonstration the equations for a revolute joint are developed and the students see how this type of constraint equation is modeled using the I-DEAS graphical user interface. Students are then given a small design project in which they are required to simulate a practical mechanism using I-DEAS. In their projects they are required to design a mechanism to meet a set of requirements, compute the joint forces associated with the mechanism, and simulate the motion of the mechanism using features within I-DEAS.

Figure 2. Four-bar Mechanism Modeled Using I-DEAS Software.

Prerequisites

In most kinematics courses the prerequisite is a first course in dynamics. In the case of the course described in this paper, the students should also be able to write computer programs.
using MATLAB, create drawings of simple mechanisms using AutoCAD, and create models and assemblies using I-DEAS. At the University of Tennessee at Martin, second semester freshmen are introduced to MATLAB and AutoCAD in a second course on Engineering Methods. They are also introduced to part and assembly modeling, and drafting using I-DEAS in a sophomore engineering design course. The differing abilities of students relative to the application of AutoCAD, MATLAB, and I-DEAS is the largest variable in the course, and the instructor must provide considerable assistance as students develop and debug their programs.

**Summary**

This paper has described an approach for introducing students in a first course on the kinematics and dynamics of machines to the fundamental concepts used in state-of-the-art multi-body simulation software. This approach is based on identifying two central concepts: 1) the difference between inverse-dynamic and dynamic analyses, and 2) constraint equations. The method for using these central concepts as a unifying thread to introduce students to classical concepts as well as modern methods is described. Students taking a course based on this method are exposed to state-of-the-art technology without sacrificing traditional course content.

**References**