As published EEWeb PULSE

WHY CAN'T JOHNNY DESIGN?

Is Engineering Education Moving In the Right Direction?

Does engineering education need an overhaul? How do educators ensure the next generation of engineers is ready to meet the complex challenges of the 21st century? Dr. Tom Lee, Chief Education Officer at Quanser, offers insights into these all-important questions in a three-part series of articles entitled, "Why Can't Johnny Design?" originally published by EEWeb PULSE Magazine.

QUANSER
INNOVATE. EDUCATE.

Part 1: The Challenges in Modern Engineering Education

These are interesting times for the engineering profession. On the surface, all the signs are there for a real engineering Renaissance. Technological magic seems to generate ever-more impressive gadgets; our cars are achieving near miraculous efficiency at affordable prices for the consumer; and modern engineering disciplines such as biomedical engineering are offering new hope to countless people who would have been functionally marginalized in decades past. So why are engineering educators so grumpy these days?

No, there is no scientific study that has tracked a "grumpiness index" among our professors but in recent years there definitely seems to be some sense of angst among the academic community. Recently at a conference of Innovation Centers for Engineering Education (ICEE), on Jeju Island in Korea, Korean professors gathered as part of a regular sequence of meetings among a network of 60 Korean engineering universities charged by the national government to close the real gap that exists between Korean engineering program and the needs of Korean industry. The industrial reality today is that Korean industry can legitimately claim the hard-fought title of "the next Japan". Certainly on the consumer side, the products of the Korean electronics and automotive industries like Samsung and Hyundai, by any measure, command brand respect approaching if not equaling legendary Japanese brands such as Sony and Toyota.

The Korean ICEE initiative is significant in that it is a focused, well-funded initiative by an engineering community that is targeting the future driven by engineering creativity and innovation rather than manufacturing quality and aggressive labor costs. The sessions at this conference focused not so much on graduating more engineers to meet increasing demands, but on the more challenging, big picture, questions about what qualities the engineer of tomorrow should embody and how does the academic community deliver the appropriate training. ICEE is currently concluding its first five year mandate from the national government of Korea and all indications are that funding will be renewed, if not increased.

Grand Challenges of Modern Engineering

The Korean ICEE experience is a case study in a regional response to the larger issue of the readiness of graduates. The most ambitious statement of the significance of the engineering requirements of the future is best articulated by the so called "Grand Challenges for Engineering". In 2008, the National Academy of Engineering in the United States presented fourteen major technological issues that vex humankind and encouraged the global engineering community to use these as a framework for assessing education, research, policy, and industrial strategies. The grand challenges are: make solar energy economical, provide energy from fusion, develop carbon sequestration methods, manage the nitrogen cycle, provide access to clean water, restore and improve urban infrastructure, advance health informatics, engineer better medicines, reverse-engineer the brain, prevent nuclear terror, secure cyberspace, enhance virtual reality, advance personalized learning, and engineer the tools of scientific discovery.

For many in the academic community these challenges have precisely articulated the significance of the work of the modern engineer but unfortunately, it also highlights how difficult it will be to prepare our students to tackle these immense challenges.

The current engineering curriculum delivered by the great majority of institutions worldwide had its genesis in the mid-twentieth century. Largely motivated by then urgent requirements of the so-called Space Race and its darker flipside, the Cold War, a large number of engineers had to be trained to meet the needs. The response from the academic community was a curriculum that one might consider to be linear -- start with mathematical and scientific foundations in the early years, progress to application courses in the middle years, then cap it off with a rigorous thesis or senior project. Additionally, the tradition of separating students into well-defined engineering disciplines (electrical, mechanical, chemical, civil, etc.) became entrenched. Although, one cannot deny that this approach was effective in that the needs of society were largely met, it left our generation with lingering memories of being completely lost for four or more years, and eventually seeing the light once we began experiencing the real world.

The mismatch for the modern context often cites two very significant issues. First, as any of us who graduated from engineering programs during this period knows, is the simple reality of keeping students motivated through the very intensive and often abstract theory-heavy gauntlet of the first years the under graduate program. "What is the significance of this calculus theorem proof in the real world?" is a question many of us asked as we struggled through the process. In some sense, this is the easier problem, as many successful techniques have emerged within the past few decades that have attempted to introduce more applications, hands-on labs, case studies, etc. to help "soften" the blow.

The trickier issue is the latter -- the disconnect between the traditional structure of the engineering disciplines and the emerging complexities of modern engineering systems. With the pace of innovation and the increasing sophistication of products and infrastructure, techniques represented by the traditional curriculum are considered by many to be out of date. Modern engineering teams are typically cross-functional, with contributions from a variety of specializations. This included technical specializations but also non-technical, involving business and human factors. The shortening project timelines also demand greater project parallelization and cross-functional tasks that simply do not map cleanly to the disciplines. Succinctly, engineers need to know more, do more, and do all this with less time and resources.

Technologically, such pressures have triggered highly innovative techniques often facilitated by modern information and digital technology. A clear testament to this trend is the academic migration of the traditional departments of Electrical Engineering (EE) to the more contemporary hybrid departments of Electrical and Computer Engineering (ECE). In another important corner of the engineering world many departments of Mechanical Engineering (ME) are starting to express themselves as Mechanical and Mechatronic Engineering (MME). This sort of trend is one of the modern responses of the academic community: to create new departments to accommodate new techniques. Is this sufficient?

The reality is this will generate a relatively small specialized group of engineers skilled in what is arguably an even more specialized and narrow field, but leaves the vast majority of our students in the framework of the traditional disciplines. A very practical example of the deficiency in this approach becomes clear in a very familiar context: modern cars.

A Practical Example: the Green Car

Virtually all consumer vehicles produced today are electronically fuel injected. More precisely, they are typically computer controlled, deploying the same semiconductor technology we use in our general purpose computers. The electronic control unit (ECU) manages the injection of fuel and ignition timing and other key parameters that influences the burning and energy release of the engine. For many cars, there will also be some computer control through the drivetrain (e.g. traction control). Within the rapid development cycles of today's auto industry, it becomes essential for the engineering team to work with the system in its totality. So in the case of the engine, you will need to consider as a minimum mechanical, chemical, and electrical characteristics. Since the engine is computer controlled, you will also need computing knowledge.

From a workflow perspective, new techniques such as Hardware-in-the-Loop (HIL) testing offers a highly accurate and rapid way of testing the performance of control systems for engines or any other computer controlled subsystem. This is accomplished in a safe, cost-effective, yet virtually realistic way of testing key components without having to physically prototype the entire system only to find a critical design flaw. This type of sophisticated simulation and control systems design will demand a system-level approach that blends techniques and knowledge from many of the specialized disciplines. Extrapolating further to green vehicle design, the same problems become greatly amplified for hybrid electric vehicles (HEV), fully electric (EV), and fuel-cell powered vehicles. For these applications, chemical engineering knowledge increases in importance as battery design, and alternate fuels become the big variables.

The Renaissance Engineer

The professors at the Korean conference expressed keen interest in techniques that introduce such multidisciplinary and system-level approaches to engineering education. In concept, it may seem simple enough to begin merging select techniques from other disciplines into programs; the actual implementation is substantially more difficult. The curriculum legacy is entangled within a large complex system of organizations and sub-organizations, often with a highly distributed structure and decision making process that ensures academic freedom but hinders coordinated transformation. Furthermore the recent focus on the research function of the engineering university also needs to be tempered to allow greater creativity and energy to revitalize the teaching function.

The Korean context was used to highlight the level of commitment and vision required to educate the modern renaissance engineer. This particular group would be the first to admit that they have only taken a baby step. But that first step is literally a "doozy". Korea is a nation of fifty million people for whom the engineering community has literally lifted the population out of war-torn poverty within a generation. For this country the revitalization of the engineering community is an issue of national priority, and government, industry, and academia seem to be in synch to effect the changes. They, however, are not alone. At another recent conference in Asia, approximately three hundred deans of engineering and other senior engineering education administrators converged on Beijing at the 2011 Global Engineering Deans Conference presented by the International Federation of Engineering Education (IFEES). Within the audience were significant contingents from North American institutions who are also anxious to learn from their global peers and trigger positive action in their respective jurisdictions.

Throughout North America and elsewhere, a generation of children has embraced robotics as a hobby and even an obsession. These same children also lose sleep at night wondering whether their world will provide sufficient food, clean water, and livable environments when they take the helm. In many ways, the scene is fundamentally different from the concerns of Western nations on whether we are training enough engineers to compete against emerging economic superpowers. The demands on the engineering community are becoming deeply personal and we witness an empowered generation ready to take on the challenges, but they can only succeed if our generation can structure our institutions and methods to guide them wisely.

Engineering deans gather in Beijing to explore avenues to transform the undergraduate curriculum.

Part 2: Reinventing the Engineering Lab

Part 1 of this series of articles surveyed the key trends and drivers that are challenging the way we train the next generation of engineers. Global political and economic dynamics, increasing complexity of modern engineering systems, and general resource challenges that vex most education institutions are some of the difficult issues that so many of us are concerned about. In Part 2, we begin looking at some of the better ideas emerging from engineering campus to take on these challenges. In particular, this article will focus on the undergraduate lab and how new technology and new thinking are triggering more effective learning.

The modernization and enrichment of the undergraduate lab experience is one of the primary elements of engineering education innovation and it is a critical part of the broader trend of introducing more and better hands-on experiences in the curriculum. Hands-on education has taken several key forms in the modern institution. Most of us are familiar with the most obvious form – the undergraduate lab –that place we all went to once a week to touch real knobs, measure noisy signals, and somehow try to connect the dots to the theory. Other prominent forms of hands-on education also include project-based learning, virtual laboratories, and the mother of all hands-on experiences, internships or co-op programs.

From Theory to the Real World

In any formal gathering of academics on the topic of teaching young engineers, the subject of hands-on education inevitably comes up. Trends towards interdisciplinary engineering, or increasing the real-world application dimension in courses, are fueling much-needed debate within academic circles and, in most cases, the vectors are pointed away from traditional, lecture-centered education.

Another driver is the availability of new technology that provides cost-effective options for education innovation. Rich information among instructors and students are possible through social networks such as Facebook or academic systems like Blackboard. Simulation and virtualization is another example of providing some level of interactive engagement with a "real enough" engineering system. Though nothing compares to actual engagement with real systems, many instructors are turning to virtual techniques to enrich the learning of topics that simply would have stayed highly theoretical and abstract. Pure simulation, in this sense, can be a very effective tool for establishing context.

The final technological influence is simply the increase in quality and sophistication of education-focused equipment that students can now access. For those of us trained in the bad old days (the 1980s for me), we have less than fond memories of aging and often inconsistent lab equipment where we had to mentally build a connection between some highly contrived experiment and a bunch of differential equations. Today's lab equipment, whether through digital evolution or simply better manufacturing practices, are more robust, more flexible, and can offer a much richer set of relevant experiences for less money.

All of these factors have established truly fertile ground for campuses to explore and implement new ideas. In the category of "This is not the lab I remember from my college days!", the modern robotics lab, based on the new generation of consumer and educational robotics platforms, has to be one of the more notable trends, and it is largely driven by the issue of student motivation.

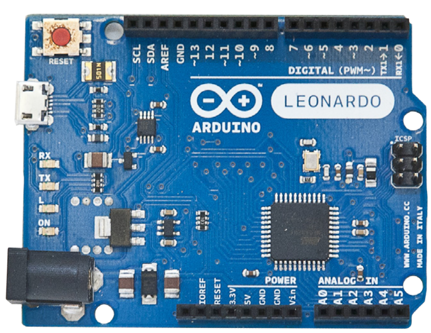
This concern about student motivation is really at the heart of a lot of heated discussion within the engineering education world. Recent statistics indicate that in the U.S. upwards of 40% of all undergrads who start science, technology, engineering, math (STEM) programs never finish. When you also consider that less than 15% of all incoming freshmen choose STEM, the final numbers of those who enter STEM careers is very low indeed.

Many institutions are investing heavily in ways to make engineering a more attractive option for young people, and interesting and even fun lab experiences are seen as part of the answer. There is no more prominent manifestation of this effort than with the explosion in consumer and educational robotics.

Fun with Robots

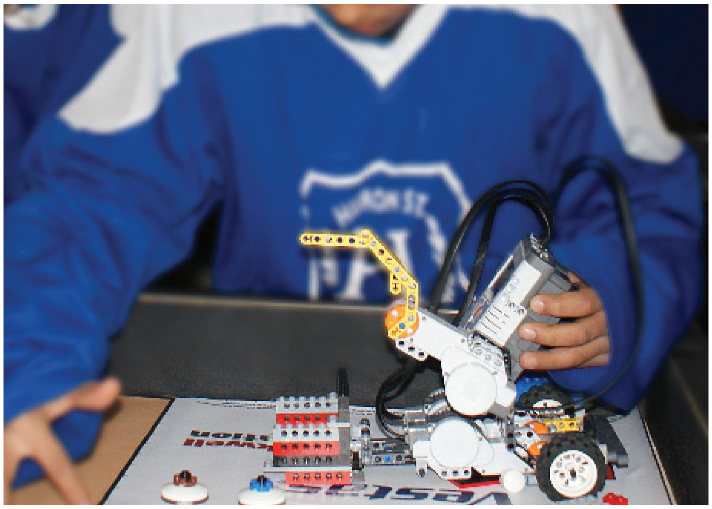
Twenty years ago, the concept of consumer robotics would be unheard of or even mocked, but today young people playing and creatively programming small autonomous robots have become a part of mainstream youth culture. Lego Mindstorms has become the most recognizable name for youth or even young kids' robotics. The FIRST (For Inspiration and Recognition of Science and Technology) organization has elevated the hobby of robotics to an international spectacle where high-school age students build complex rover-style robots that would have been considered research-grade a generation ago. The open platform Arduino has become the grown-up hobby roboticist's platform of choice and has spawned countless weird and wonderful, crawling, flying, and dancing mechatronic critters.

This trend has not gone unnoticed by universities and indeed, Arduino or Lego-based projects (Figs. 2,3) and labs are a very popular option to add some sizzle to the undergraduate lab. In fact, an interesting observation is the introduction of such robots into the freshman curriculum. At the University of New Mexico in Albuquerque, the school of engineering now



Thanks to its flexible and easy-to-use hardware and software, Ardunio became a platform of choice for hobbyists.

requires students to program an Arduino platform autonomous ground vehicle to navigate a complex course. The goal was to bring some life to a notoriously hated course, "Introduction to C Programming", by adding an engaging project. By all measures, this has been a successful initiative and the course continues to include such a project. In fact, similar additions of robotics labs into existing courses are very common at many institutions.



Grade 5 kids can easily program Lego Mindstorms and compete in FIRST Lego League.

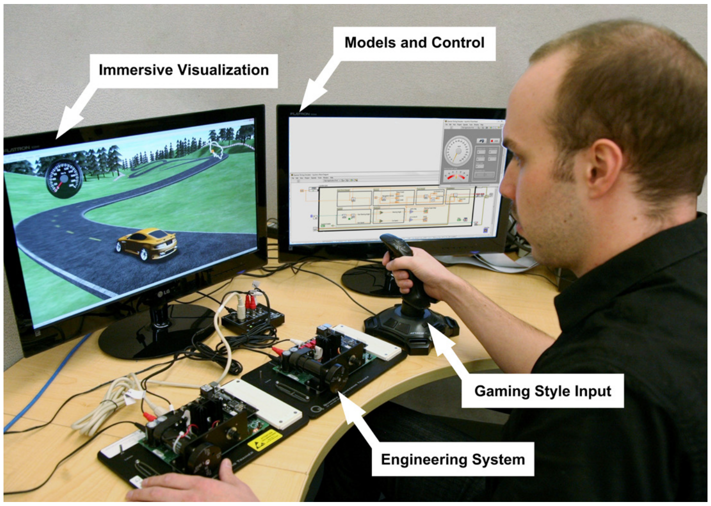
The robotics lab project also recognizes the broadening of the role of computers in engineering. The traditional curriculum viewed computing as a platform to program algorithms to manipulate data and this is why we have always taught sorting algorithms and textbook-loads of numerical methods for seemingly every type of math problem. In a recent meeting of the Canadian Field Robotics Network held at McGill University in Montreal, Professor Gregory Dudek remarked that he believed that the primary goal of all computer scientists should be to move computing off the desktop and onto mobile platforms – that the days of the "Hello World!" programming course was over.

This is setting up an interesting situation. We now have a generation of kids coming into college with practical and sometimes advanced knowledge in robotics and real-time computing. They may not know the formalisms or all of the proper terminology, but they know how to make machines literally dance and sing. On the other hand, you have the community of engineering educators struggling to make sense of a curriculum that dates back to the 1950's (or the 1750s if you consider the math sequence!). In a perfect world, the new empowered generation would represent an opportunity to enrich the curriculum and build on top of this new enlarged skill set. In reality, however, it can be problematic as the inherent conflict between hobby robotics and large majority of the traditional curriculum are simply too great. Tradition focuses on rigor and theory, while dancing robots often reward cleverness and trial and error.

Application-Centered Labs for Control Systems

Recently, at the 2012 American Society for Engineering Education (ASEE) Annual Conference in San Antonio, Texas, I, representing Quanser, the control systems lab solutions company, presented a new lab concept that was developed in collaboration with the University of Toronto. Called the Quanser Driving Simulator (QDS), the concept was the integration of a strong application context on a more traditional undergraduate lab concept in control systems (see figures on page 5). This approach maintains the rigor of the traditional theory but layers a more intuitive and engaging context.

In contrast, an Arduino or Lego-based project basically steps around the theory and there is typically no consideration of the system dynamics, deep analysis, or optimal design. That world tends to work on trial and error and heuristics. Ultimately, motivation, hard work, diligence and creativity are the key qualities that are nurtured – all very important but, in the minds of many, insufficient to deal with the complexity of real systems. The Driving Simulator attempts to introduce the motivation and the other soft qualities via the application that remains squarely within the framework of a traditional lab.



System components for the Quanser Driving Simulation lab concept.

The approach takes a very typical undergraduate control lab system – two DC servo motors – and maps the control and operation of these motors to a car driving simulation. So rather than students exploring the impact of a parameter on the damping response of a naked motor, the application clearly shows how such variations can influence a more intuitive response in a car. To increase the realism, the lab also has within the control loop, an option for manual driving of the car via a game controller, and 3D virtual reality-style visualization. In real time, CAD-based renditions the car, road, and terrain provide a very video-game like experience.

The manual mode is typically used early in the lab sequence to motivate the students. There is also an automatic driver mode where formal, rigorous concepts of control can be addressed.

This comparison to the video is actually quite important. The system was intentionally designed to appeal to the incoming video-game generation. The belief is, of course, having some fun with your lab early on is not entirely a bad thing and will motivate you to work through the more challenging material. The critical difference is that there is actual plant Hardware-in-the-Loop (the motors) with sufficient fidelity in the models to map well to real engineering, and in particular, Hardware-in-the-Loop (HIL) testing of control systems.

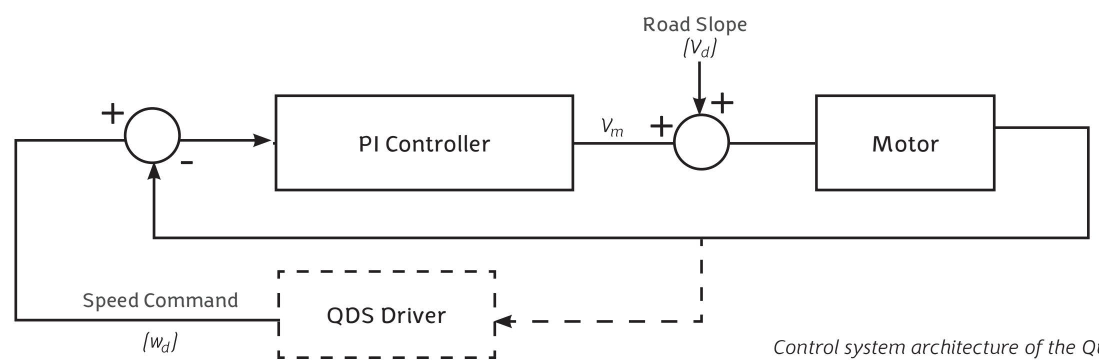
Pioneered by the aerospace and auto industries for improving controller design, it emphasizes a simulation and modeling approach, but one where there is also some system hardware (a plant) within the control loop. With HIL, engineers can achieve higher fidelity simulation runs that still embody real world effects, non-linearity, and noise that only true hardware can provide. The driving simulation lab's rendition of HIL is a reasonable replication of an important modern industrial workflow.

In terms of control systems theory, the Driving Simulator successfully navigates the key concepts of modeling, system characterization, and compensator design. The Toronto students, based on course grades, had similar or better comprehension than with traditional labs. Anecdotally, the instructors felt that there was a distinct increase in the level of motivation and interest in the course. Basic observations such as students wanting to remain in the lab long past designated hours, or developing very creative variations of the driver model tied to the controller model (e.g. a sober driver vs. a drunk driver) are examples of student behavior showing very positive changes. The University of Toronto has recently announced that it will increase the use of this lab concept for their control course.

In the end these two lab concepts should instill some optimism within the engineering community. Dr. Jacob Apkarian, Founder and CTO of Quanser, recently remarked that curriculum transformation is more like moving a cemetery rather than moving a house. The decades, if not centuries, of tradition is in fundamental conflict first with the changing youth culture and demographics of today's students, but also with the changing realities and expectations of modern industry.

The good news, as was demonstrated by the case studies highlighted here, is the fact that it was relatively easy to do. In the case of consumer robotics, the information is all out there and the components are cheap. The main challenge is organizing the class and providing help to the students as required. And ultimately, help will likely come from other students or on-line anyways. With the Driving Simulator, the transformation from abstract to engaging was, for all intents and purposes, through clever use of software. All other components were off-the-shelf and well-established. It is true that this approach is significantly more expensive, but relatively speaking, even advanced lab hardware experiences the general reduction in cost and performance improvements that consumer tech products do. The slope may not be Moore's Law, but there is definitely a general tendency in the right direction.

For many, including myself, the more rigorous approach embodied by the Driver Simulator merits serious consideration as it is not tied to a consumer trend and it maintains a healthy respect for going beyond trial and error in design. Techniques such as HIL are showing that you cannot simply "rule of thumb" your way to designing a hybrid powertrain. The motivation and fun side, however, should not be dismissed. When we were in college we joked about awful courses because we thought that's the way things were supposed to be and things will never change. But with declining enrollment and pressing societal challenges that need more and better engineers, any effective way of retaining bright students is well worth the effort.



Control system architecture of the Quanser Driving Simulator.

Part 3: Doing the Math

It has been almost 25 years since I received my Bachelor's degree in engineering from the University of Waterloo. My classmates and I are now a little older, a little heavier, a lot greyer, and hopefully a little wiser. On those occasions when I do meet up with my friends, the conversation inevitably diverts to recollections of our experiences as students. Among the tales of dorm parties, all-night cram sessions, impossible exams, and unintelligible TAs, as sure as Newton's Second Law, the topic of math comes up. Typically it comes up in context of those aspects of the college experience that people hated.

This, of course, is an overly broad generalization but I have observed that more often than not, graduates cite their math courses as having been some of their least favorite or least useful for their careers. In the US, the Accreditation Board for Engineering and Technology (ABET), requires that an accredited program must instill in its graduates, "an ability to apply knowledge of mathematics, science, and engineering". This criterion is the first among the eleven key criteria demanded by ABET for accreditation. In the US and Canada, the undergraduate engineering student takes, nominally, three calculus courses, an algebra course, likely a statistics and probability course and, as electives, partial differential equations, numerical methods, and countless courses where more than half of the course is essentially mathematical.

So here is the essential conflict. The system believes that math is king and students think that math sucks. In 2008, I wrote about a new discussion group that emerged in the early days of Facebook's history. The group was called "Every time I walk into math class a little part of me dies", and its avatar had in bold letters "Math Sucks". This group very quickly grew to over 12,000 members who shared a common interest – the fear, loathing, and ultimately hatred of math.

Of course being hated is not the same as being useless. I hate turnips but I am fully aware of their nutritional value. Having said that, some part of these sentiments is shared with those who are more experienced and knowledgeable. The comments that I have received from readers of my previous articles often cite the disconnect between the theoretical emphasis of conventional engineering education and what they believe are the core skills of the successful modern engineer. Often the criticisms seem to center on the notion that real life is not an idealized closed linear system and a broad range of technical and human skills are necessary.

Calculus as a Pump Not a Filter

In 1987, the Mathematical Association of America (MAA) released a landmark proceedings entitled Calculus for a New Century: A Pump, Not a Filter. The analogy was that for too long, institutions have been using math courses as some metric of student ability for engineering and the physical sciences – those who do well in math must be the smartest and therefore will likely become the best engineers. You can debate whether such "boot camp" approaches are good or bad but this is missing the point. Math does have a contributing role to engineering practice except, in my opinion, the system has yet to figure out how best to present the case. In other words, math can be a pump to enable greater performance among students.

Fundamentally, the principal role of mathematics, and the reason why it appears first on the list of ABET requirements, is its ability to serve as models of physical and other systems of interest to engineers.

In engineering, students are immersed in calculus because rates of change (derivatives) are very good at mapping to changes in physical, measurable variables such as energy, position, concentrations, etc. The topic of differential equations ultimately coalesce the foundational techniques of limits, differentiation, and integration to formulate, solve, and apply differential equations in a modeling context.

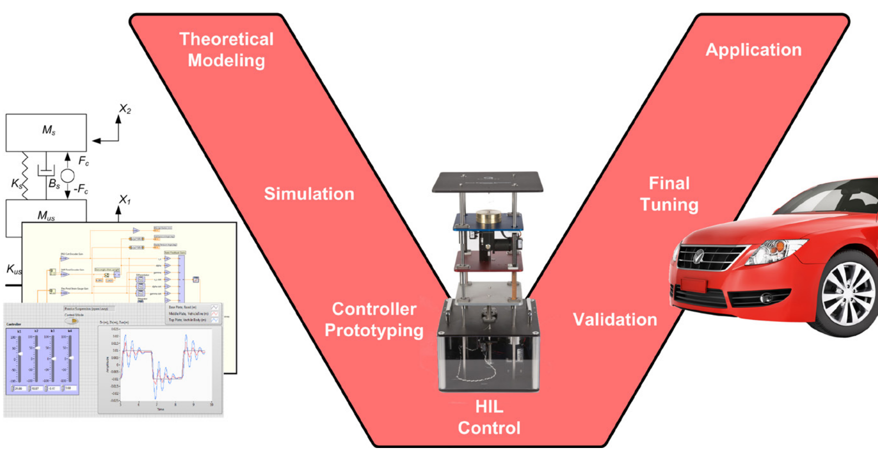
Model-Based Design in Industry

From a practical and industrial perspective, the theoretical tools of modeling are supposed to offer the engineer the ability to predict possible behavior of designs and provide guiding information for sorting through large complex sets of possible parameter values. In an ideal engineering workflow, mathematical modeling, analysis, and with the advent of powerful modern computing tools, simulation, precedes development of prototypes and testing to ensure that as much iteration is done in a virtual or theoretical environment where revision and iteration is significantly cheaper and safer. The net result is supposed to be overall reduced time in the development and refinement of designs (i.e. cost savings) and greater performance (i.e. better designs).

In complex systems such as cars and aircraft, the notion of building complete prototype vehicles based on best guesses on the designs of subsystems is no longer a feasible workflow, as design timelines shrink and competitive pressures increase to unprecedented levels. Such industries have been steadily refining the techniques of model-based design.

The figure on this page shows a common representation of the process. In this case, the overall process is for the development of a complex automotive control system (e.g. active suspension, ABS, fuel injection, etc.). The "V" shape outlines the two broad groupings of phases connected by a step called Hardware in the Loop (HIL) testing. The left arm of the V is the sub-process where a sound mathematical foundation is critical. The right arm of the V is concerned with how effectively the theoretical outcomes of the left side is tuned and optimized for the final application and the real world. HIL testing is the systematic evaluation of subsystems where some configuration of models and actual hardware are connected to assess design performance with successively more realistic hardware configurations.

When model-based design works, the preparatory work, largely framed by mathematics, allows you to predetermine likely parameter values and potential hazards. This ensures testing on hardware starts from much better initial estimates and the theoretical predictions guide how refinements and tuning can be done. Consequently, you can achieve optimal designs faster.



The "V" diagram representation of Model-based design for automotive control systems.

All major automotive companies, as well as others engaged in advanced engineering (e.g. aerospace, biomedical, robotics, high-precision machinery, etc.), have signed on to this technique for the simple reason that stakes are getting very high and engineers are seeking greater sophistication in their methods to deal with increasing complexity. In 2009, Toyota initiated the very well-publicized recall of 5.2 million vehicles due to reported unexpected acceleration. In the end, the cause was deemed to be a purely mechanical problem of sticky pedals or poorly placed mats. The nightmare scenario for Toyota was, however, that it was a problem with the control electronics for the powertrain – e.g. the engine control unit (ECU) improperly injects more fuel at the wrong time due to a software bug. At the time of the recall, the company deployed significant resources to quickly determine the source of the problem, but the specter of a potential controller bug was driving a quick and sweeping response from Toyota and in the end triggered a very expensive recall.

The fix of a controller program bug may seem simple – i.e. fix the bug and reprogram the controller – but the actual problem could also be systemic, reflecting a deficiency in their modeling, controller design, programming, and countless other soft variables that influence a complex programming activity. In the end, from a business perspective, a company like Toyota, which built its market position on quality and engineering excellence, cannot afford to have its reputation tarnished by suspicions of buggy or non-robust processes.

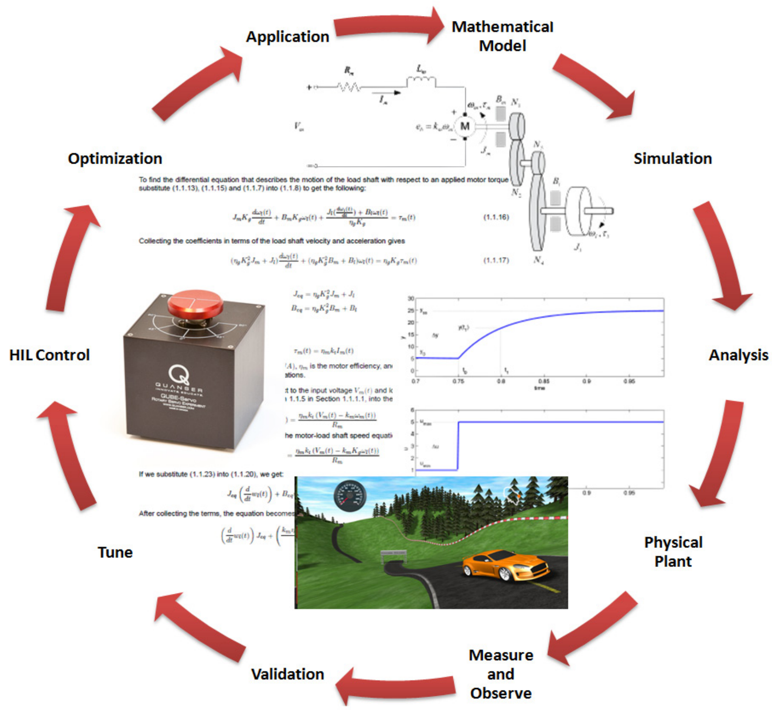
Implications for engineering education

This context of this article is math. The example of model-based design is not intended to make the point that math can somehow fix all of these complex problems. The main point is that tools such as math and modeling can no longer be considered some isolated set of techniques that is isolated to a university. Global business drivers are demanding a much broader range of solutions to be brought to bear on a complex design problem. The sophistication of modern engineering is not so much these specific solutions and techniques but the way they have connected and integrated into contemporary processes as in the one example of model-based design.

My company, Quanser, is in the business of developing hands-on lab systems that better reflect modern engineering workflows. We are currently working to improve undergraduate lab exercises, with a particular focus on control engineering. The greatest challenge we face is reconciling the traditional compartmentalization of the curriculum with the more integrated nature of engineering processes today. Within the context of control engineering, as early as 2000, key leaders in the academic and industrial control community began prescribing remedies for resolving the educational disconnects. Richard Murray of Caltech struck a panel of global authorities in control and summarized their recommendations in "Future Directions in Control in an Information-Rich World" (IEEE Control Systems Magazine, April 2003.

> *The community must continue to unify and compact the knowledge base by integrating materials and frameworks from the past 40 years ... It is also important that these courses emphasize the principles of control rather than simply providing tools that can be used in a given domain.*
>
> *(Murray, Åström, Boyd, Brockett, Stein 2003)*

This emphasis on the connectedness of the theoretical and the practical application define both the methodology that we infuse into new lab concepts that we are currently developing. In Part 2 of this series, the Quanser Driver Simulator (QDS) was discussed. This approach marries control theory with a visual automotive driving simulation within the framework of an HIL control loop. The educational framework that supports this concept is very much in response to the type of recommendations that Murray and others have offered. The figure below illustrates the ideal integrative process that QDS aims for.



The Role of Mathematical Software

The various laws of physics tell us that if we are to add stuff to a closed system then something else needs to give. So the practical question of what has to give in terms of the traditional theoretical curriculum in response to new concepts and techniques within a finite course time limit is critical. In the context of the math, one key part of the answer is the emergence of software that has been adopted by industry and has also begun influencing education.

The first group is composed of general system-level modeling tools. Examples include Simulink® (based on the math language MATLAB®) from The Mathworks, and LabVIEW® from National Instruments. Both are general platforms where you can define complex system models of physical systems via direct math or logical programming within a block diagram framework. Furthermore, they offer a broad range of analytical tools for control analysis and design, and ultimately HIL control implementation via a series of embedded system programming tools. Both of these platforms are in wide deployment in industry, though they have particular respective advantages. They have also become part of the fundamental toolset that most engineering education programs introduce to students, either to aid in the efficiency of certain math calculations or to better reflect industrial practice.

The second group includes general-purpose symbolic math tools including Maple™ from Maplesoft, and Mathematica® from Wolfram Research. These differ from systems like MATLAB in that they are able to directly represent and compute with algebraic equations without resorting to floating point values. These programs can directly differentiate the function x2+sin(a x) symbolically to 2x+a cos(a x) instead of using a numerical algorithm producing a table of floating point numbers. Such packages have had limited adoption in industry but have had sweeping adoption within the core math courses in universities. In fact, the calculus reform of the 1990's was intimately related to the refinement and proliferation of these packages because of their ability to preserve underlying mathematical meaning but remove the nitpicky detail of the actual steps of algebra.

A third, very interesting group has now emerged that seems to show the potential for integrating the general system approach to modeling with theoretical rigor all within the context of a practical, industry-oriented engineering workflow. Loosely grouped as "physical modeling" systems, packages such as Maplesoft's MapleSim™(derived from Maple) and Dassault System's Dymola® (based on the open modeling language Modelica) offer a particular form of block diagram approach to modeling that is more suitable for expressing complex systems with a variety of component types, with the mathematical model equations remaining intact for more advanced analysis. Furthermore, these tools are compatible with industry-standard software such as Simulink and LabVIEW, allowing them to bring a more mathematical and rigorous approach to practical control system development and HIL testing. This class of systems has begun deploying in automotive and other sectors in key engineering communities, such as Japan, where engineers are betting that a more thorough modeling framework may deliver more comprehensive testing and preventing catastrophic errors in control system design.

In education, however, this combination offers a more interesting opportunity. Conceptually it now provides a toolchain where the mathematical approach that so dominates our classrooms and textbooks have a well-defined and accepted place in industrial practice. Although symbolic math tools have been in place for over two decades in the math curriculum, it still struggled to find meaning within the engineering application courses encountered in the later years.

At companies like Quanser, this is welcome news as a rigorous treatment of the modeling phase of design has always played prominently within our own R&D workflow. Many of Quanser's most strategic products, such as the quadrotor UAV and the haptics devices, contain highly complex mechanical dynamics that require mathematical techniques to refine the required controllers. This modern generation of analytical tools can not only make our workflow more effective, but these same techniques can then be mirrored within the lab exercises that we develop based on these devices, and be introduced in a meaningful way into academic applications.

What's Next?

In all, universities and industries have taken modest steps at best in reconciling the disconnect between traditional curriculum and modern practice. Though it is clear that the technology pieces are likely robust enough for use by students, the inertia of centuries of academic tradition is difficult to overcome even with the best of ideas and tools. Recently, I was invited by the American Society for Engineering Education to serve on a committee to develop recommendations for a more comprehensive strategy for closing the industry-university gap. In Part 1 of this series, I wrote about the efforts being undertaken in Korea and China to modernize the curriculum. Although all of these initiatives are tackling a much larger set of problems than just math, math continues to stand out as that thing that everyone probably believes is a good thing but we have not figured out how best to present it in a motivating and relevant way. But the discussions and debates have at least started and the recent developments of innovative tools and techniques are beginning to put the pieces of the grand puzzle together.

But something tells me that there may be deeper and even cultural issues to all of this. Math-phobia, at least, in North America, is a chronic problem. I often cringe when a well-educated, articulate person openly boasts in public that he or she stinks at math. When was the last time you saw someone proudly proclaim that he was illiterate? As with any human endeavor, there are those who are better at math than others. But there are those who write poetry better than others, or can build a cabinet better than others, or can cook paella better than others, or can read Latin better than others, or can hit baseballs better than others. Ultimately, the specific techniques of calculus and algebra are not beyond the capacities of the vast majority of our engineering students if we can make the need for engineering math proficiency as obvious as making a great paella. I am hoping that concerted efforts in observing and listening to industry, as well as collectively pausing to smell the paella… that is, reflecting on the empowering and creative nature of mathematics and modeling and moving away from the rigid compartmentalized framework of the traditional curriculum, are a few more pieces of the puzzle.

About the Author

Dr. Lee has been an active contributor in the global engineering and control systems community for over twenty years. As Chief Education Officer at Quanser, a leader in real-time control and mechatronics solutions for education, research, and industry, Dr. Lee develops and implements the company's strategy for enriching and increasing the educational effectiveness of technology in the modern engineering education context. Prior to his appointment at Quanser, Dr. Lee was Vice President of Applications Engineering at Maplesoft, creators of the renowned Maple mathematical software system. In that capacity, he helped the company transform the mathematical technology to a complete engineering modeling and simulation solution. He also serves as an Adjunct Professor of Systems Design Engineering at the University of Waterloo, noted for its leadership in engineering, computer science, and mathematics. Dr. Lee earned his Ph.D. in Mechanical Engineering at the University of Waterloo, and his M.A.Sc. and B.A.Sc. in Systems Design Engineering at the University of Waterloo. He has published numerous papers and is a frequent invited speaker in the areas of engineering education, engineering modeling and simulation, and engineering computation.