A Multidisciplinary Model for Using Robotics in Engineering Education

Jerry B. Weinberg, George L. Engel, Keqin Gu, Cem S. Karacal,
Scott R. Smith, William W. White, and Xudong W. Yu
Southern Illinois University - Edwardsville

Abstract

The use of robotics to provide hands-on instruction across the various disciplines of engineering and computer science is no longer the prohibitively expensive proposition it once was. With the emergence of inexpensive robot kits that encompass a background in electrical engineering, mechanical engineering, industrial engineering, and computer science, robotics can now play a central role in the education of students in these disciplines. A critical obstacle to this goal, however, is the lack of familiarity that students in each discipline have for the other fields of study, making a thorough understanding of overall robotics design principles quite difficult. This paper presents a model for multidisciplinary cooperation that alleviates this problem and elevates robotics to a potentially pivotal position in engineering education.

I. Introduction

Robotics provides a comprehensive view of an integrated, fully engineered system. It affords a view of information processing from the microprocessor level up through the application software, and it illustrates the connection between mechanical, electrical, and computing components. Because of its multidisciplinary nature, the study of robotics in the classroom can be a valuable tool for the practical, hands-on application of concepts across various engineering and science topics.[1] Furthermore, the curriculum in any specific area of study tends to narrowly focus students on that area, whereas real-world complex systems tend to integrate electrical, mechanical, and computing components. The study of robotics provides a medium for students to experience this integration and to see the interaction between the various types of systems.

Its multidisciplinary nature has also relegated the study of robotics to larger research universities and private industrial research groups whose members have had the full range of prerequisite knowledge to engineer such complex systems. Pre-constructed industrial robots could be purchased, but their exorbitant prices made them cost prohibitive to the more modest budgets of smaller educational institutions. With the emergence of inexpensive computational components, robot platforms have become more accessible to such smaller programs.

More importantly, these platforms have made the area of robotics accessible by removing the need to have a background in electrical engineering, mechanical engineering, and computer science simultaneously. Platforms such as the Handyboard and the LEGO RCX[2] have managed to allow users to cross the *threshold of indignation*, which is “the maximal behavioral component that we are willing to make to get a task done.”[3] If end users perceive that their efforts must go beyond this point, a new tool will not succeed in the consumer market, no matter how good or interesting the manufacturer believes it to be. These robot platforms provide users with simple techniques for connecting sensors and motors, as well as straightforward methods for programming the controllers that manage those components. The LEGO RCX platform is particularly interesting in this respect. From the electrical engineering perspective, it provides a variety of pre-constructed sensors as well as motors. From the mechanical engineering perspective, robot bodies can be constructed from the simple building blocks of standard and specialized LEGO parts, which include gears, axles, and hinges. Finally, from the computer science perspective, there are a variety of programming languages available that support input from sensors and output to motors, including numerous languages that require no previous programming background.[4]

With the development of these inexpensive and accessible platforms, robotics projects provide an opportunity to directly interact with technology, as well as an opportunity to design and implement the various concepts that they embrace. Seymour Papert termed this style of learning "constructionism."[5] This approach to teaching creates an active learning environment in which students can explore a significant design area, make hypotheses about how things work, and conduct experiments to validate their assumptions.[5,6] Robotics projects are becoming a valuable pedagogical tool that is being used to teach a wide range of advanced concepts.[7,8,9,10]

Without formal guidance, however, students in a particular discipline could be overwhelmed by designs that prove to be impractical from the perspective of other disciplines. Some courses overcome this problem by providing the students with those elements of the project that are not in the designated area of study, e.g., giving computer science students a specific mechanical platform and/or sensor configuration.[11] Other courses use a structured exercise approach, in which students are given a number of exercises to familiarize them with the relevant concepts of other disciplines.[12] For this approach to be effective, instructors need to have sufficient background knowledge to formulate effective learning exercises, e.g., an understanding of mechanical gears and structures, electronic sensor limitations, as well as basic algorithmic design and multitasking.

To address this need for cross-disciplinary knowledge, we formed a Multidisciplinary Project Action Group (MPAG), which includes faculty members from Computer Science, Electrical & Computer Engineering, Industrial Engineering, and Mechanical Engineering. The MPAG provides a forum and basis for sharing expertise across the disciplines, with the goal of helping to form learning activities that are effective for students in each discipline. Consequently, students in mechanical engineering can learn enough about structured programming principles, behavior-based robotic control, and multitasking to successfully implement a control program. Conversely, computer science students can learn enough about sensor processing, gearing, and transformation power to successfully design a physical robot structure. In essence, the MPAG is a cross-functional design team for educational experiences.

II. Robotics Multidisciplinary Project Action Group

The Robotics MPAG consists of members from various disciplines: Computer Science, Electrical Engineering, Mechanical Engineering, and Industrial Engineering. The group's main goal is to share expertise for the express purpose of using inexpensive robotics platforms for teaching engineering and computer science concepts. The framework for sharing this expertise includes exercise design discussions, the hiring of student assistants between the areas, demonstrations, and guest lecturing.

Members of the group create robotics project modules that encompass concepts to be mastered in structured exercises for courses in their respective areas. These modules provide a basis of concepts and technical vocabulary for design discussions between the members. Through these design sessions, the technical concepts of one discipline are translated into materials and exercises at a level that students in a complementary discipline can understand. Members work together to adapt and expand the modules in order to make the content accessible to students outside of the specific area of expertise. Essentially, the instructors become students who gain a fundamental understanding of the relevant aspects of the other disciplines via the sharing of these modules.

Hiring student assistants from each other's discipline provides additional opportunities for the cross-fertilization of expertise. For example, having a mechanical engineering graduate student assist in the administration of the robotics equipment and lab exercises for courses in computer science provides an opportunity for knowledge to be exchanged between the two areas. The student assistant provides a readily accessible resource for the computer science students regarding questions of a mechanical nature. Furthermore, mechanical engineering students can fine-tune structured exercises by trying them out first and then suggesting possible improvements.

Finally, expertise is directly shared across disciplines by means of guest lecturing in courses and having students in one discipline demonstrate their projects to students in another. This latter approach provides a good opportunity for students to practice presenting technical concepts to an audience from an alternative area of expertise, an important real-world skill (as evidenced by the need for the MPAG itself).

Area	Course	Concepts Emphasized	Concepts Shared
Computer Science	Artificial Intelligence	Embedded agents, deliberative/ reactive robot control, planning, multitasking	Subsumption architecture, search strategies, multitasking, cross-compiling, multiplexing
Mechanical Engineering	Robotics Mechatronics	Sensor processing, logic circuits, real-time processing, actuators, analog/digital conversion, electro-mechanical system integration	Differential motion, gearing, translation motion
Industrial Engineering	Engineering Problem Solving	Problem formulation, structural design, algorithmic design, search strategies, gearing, drive train	Problem analysis and definition, integrated system design
Electrical & Computer Engineering	Senior Project	Signal processing, robotic system design, and project management, analog/digital conversion	Sensor characteristics, robotic system integration, robot navigation strategies

Table 1: A sample of concepts emphasized and shared

Prior to this effort, individual members of the MPAG felt that they did not possess the necessary expertise to assign robotics exercises in their courses. The group was formed in Fall 1999. Beginning with the Artificial Intelligence course in Spring 2000, robotics projects have been included in every MPAG member's area of study. Table 1 demonstrates the types of concepts emphasized in each area of study, as well as the concepts that have been shared with other areas of study.

II.A. Artificial Intelligence – Computer Science

The Artificial Intelligence course emphasizes the development of agents as a framework for creating intelligent systems. The robotic platforms provide an opportunity for students in computer science and computer engineering to design and create autonomous agents that are embedded in the physical world. Students must design an agent's mechanical structure, sensor input, and computation control to deal with the challenges of being in a physical world.[13]

Students in this course have the opportunity to explore a variety of advanced concepts, including intelligent agent design, deliberative/reactive/hybrid robot control, managing uncertainty, and planning. In the process, the students are exposed to other concepts, such as multitasking, cross-compiling, multiplexing, sensor processing, infrared communication, gearing, and differential motion.

Early in the course, students are assigned several structured exercises designed to introduce them to the robot platform, various mechanical techniques, and the concept of behavioral programming, all of which might be used as part of a robot strategy in the larger project assigned later in the course. These exercises include the development of such robot behaviors as path following, obstacle avoidance, and searching.

The design project this past semester was a predator-prey competition. Student teams were divided into predators and prey. The goal of each prey robot was to traverse an arena while avoiding being tagged by a predator, and the goal of each predator robot was to seek out and tag the prey (See Figure 1). The robot control program was required to use the behavior-based "subsumption network" in which individual robot behaviors are prioritized and activated based upon sensor input.[14,15]

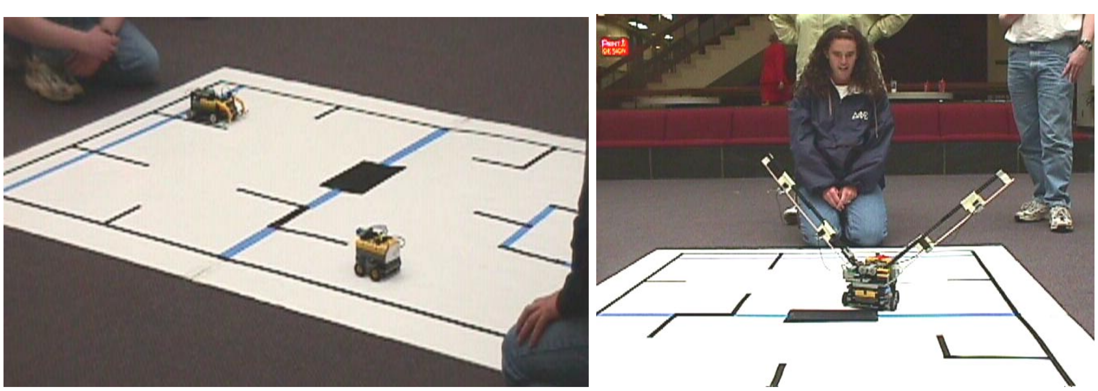

Figure 1: The predator-prey competition was played in an arena consisting of black boundaries that represented walls and blue boundaries that representing scoring zones.

II.B. Mechanical Engineering: Robotics and Mechatronics

The Robotics course studies robot structure, kinematics, dynamics, trajectory planning, and automatic control. The robot systems enhance learning by providing the hands-on experience of building robot structures and exploring kinematics.

Projects in this course vary from semester to semester. One semester, the course focused on the development of robotic arms, exploring a wide variety of arm mechanisms, each incorporating various degrees of freedom. Students were given the opportunity to build a variety of mechanical components and to attach sensors and actuators that were controlled through the computational component of the LEGO RCX.

The Mechatronics course studies the components and integration of mechatronics systems consisting of sensors, actuators, and mechanical, electrical, and computational elements. All of these can be physically realized using the LEGO building blocks. The robot controllers especially provide an ideal platform for learning real-time programming with input from the sensors and command output to the actuators.

To familiarize mechanical engineering students with the robotics system and the programming language, the students were assigned exercises to build a robot with one of two configurations: Bumbot or Linebot. The Bumbot configuration recognizes an obstacle using tactile sensors, performs a reversing motion, and then changes its course of motion. The Linebot configuration uses light sensors to “visually” recognize a marked line and follows its path. Through these exercises, the students gain an appreciation of the roles of sensors, actuators, feedback, and real-time programming.

With the experience gained from the first project, students are assigned a robot design project involving the interaction between two robots engaged in a game of tag, with the object of tagging each other on the front bumper as many times as possible. Strategies included both aggression and avoidance. The robots were required to stay within a marked boundary, using visual functions achieved by means of sensors and programming, and they were required to make certain gestures in response to tagging or being tagged. This game was competitive, with substantial opportunities for creativity, and it required a substantial effort in programming and configuration design.

Through these projects, the students gained a first-hand knowledge of a mechatronics system, especially the roles played by the key elements of such a system: sensors, actuators, control and logic units, interfaces, and real-time programming. In the class survey conducted at the end of the semester, the vast majority of students expressed the feeling that the projects greatly enhanced their learning experience.

II.C. Industrial Engineering: Engineering Problem Solving

This course, a freshman-level, general education course targeted to pre-engineering students with no expectations of prerequisite knowledge, focuses on critical thinking and problem solving methods in the context of various engineering disciplines and computer science. By participating in a robotics project, students in this course are able to explore a variety of topics in each discipline.

Some students are assigned projects using the LEGO robot kits. Examples of projects are Robo-Sumo Wrestling and Hide-and-Seek. Students work in teams of four or five, designing a robot and the program it needs to accomplish particular tasks. These projects help to introduce basic design concepts early in the curriculum and to introduce students to certain advanced concepts in each of the disciplines. The students are required to go through a problem analysis and definition process, stating the assumptions and constraints that they are applying to their designs. The teams draft a physical design and create an algorithmic solution to the robot control (see Figure 2). Preliminary rounds are held for teams to test their initial design hypotheses.

In this course, students obtain practical experience with concepts that include integrated system design, motion, gearing, structural strength, center of gravity, programming, algorithmic design, multitasking, sensor processing, and team development.

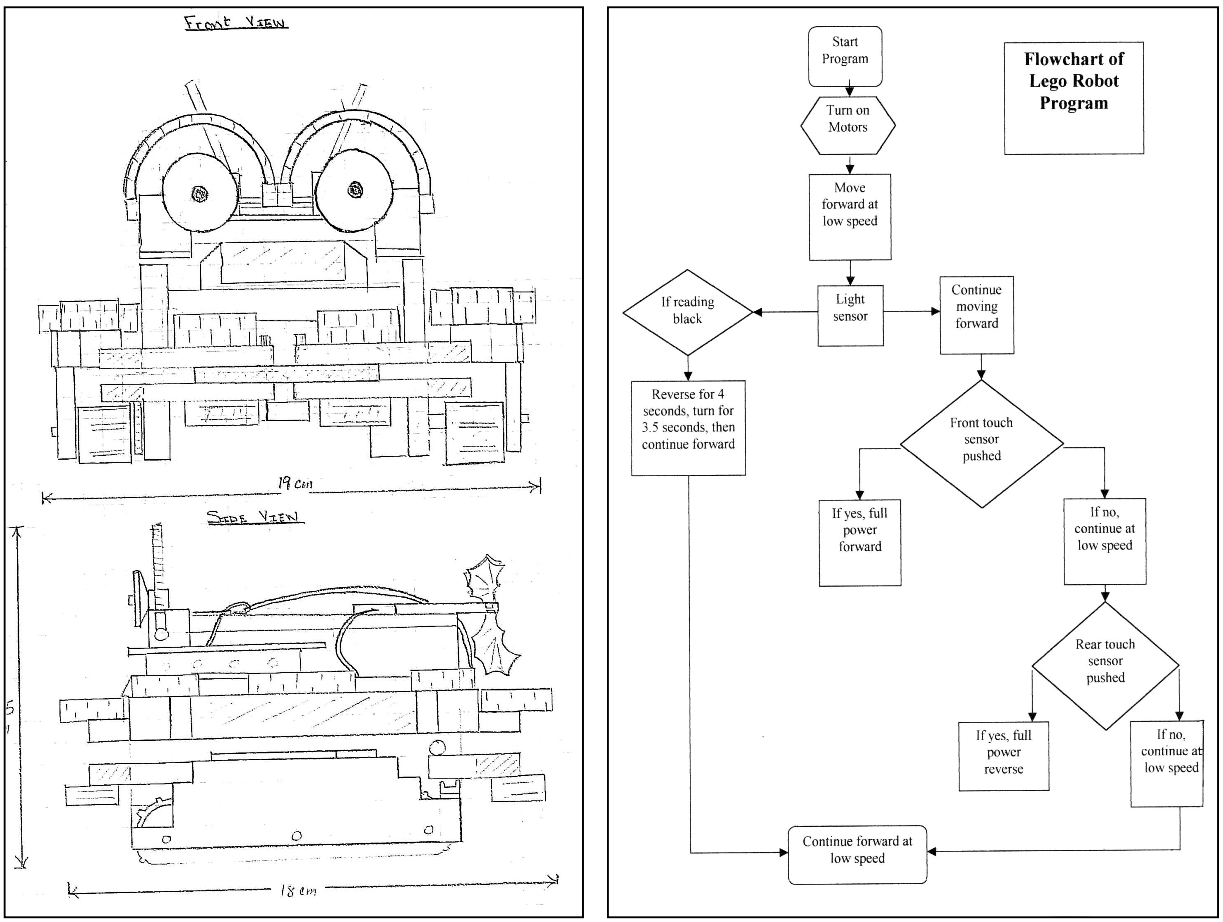

Figure 2: Example of student's structural design and robot control flow

II.D. Electrical & Computer Engineering: Senior Projects

The Senior Project courses in Computer Science and Electrical & Computer Engineering provide a type of capstone project. Robotics projects engender a full range of activities in both of these disciplines. To build a robot, students must analyze the type of environment that the robot will encounter, determine what sensor inputs are necessary to recognize different conditions, decide what motor responses will be necessary, and design the overall robot program control.

A current project in these courses involves participation in a regional IEEE robotics competition requiring the study of the difficult problem of robot navigation. As part of this project, students are designing sensors that are non-standard to the LEGO RCX, such as a directional "compass" sensor. In addition, to overcome the severely limited number of sensor inputs on the LEGO RCX, the students are designing multiplexers to provide additional sensor ports.

III. Future Work

The MPAG approach has provided a forum for a straightforward exchange of expertise that has allowed faculty in a variety of disciplines to successfully introduce robotics projects in their respective areas of study. Furthermore, the approach has provided each instructor with the knowledge necessary to provide students with a context with which they might structure their active learning experiences.

The MPAG is currently developing a cross-disciplinary course in engineering design and robotics. A main goal of this course is to bring students from different areas of study together into multidisciplinary design teams. In much the same way that the MPAG has had success in sharing knowledge, these cross-functional teams will provide students with an opportunity to collaborate with others in complementary areas of expertise.

In addition, members of the MPAG are researching the application of interactive graphics to develop a computer-aided design tool with which students will be able to design and program "virtual" robots. Once these models are satisfactorily developed, their specifications will then be used to produce real physical robots, with programs that are derived from the graphical manipulations that were recorded during the design phase. While simplifying the algorithmic process for non-programmers, the development of this interface could also provide a mechanism for automating the sharing of cross-disciplinary engineering expertise. Limited expert systems could be developed that would ensure adherence to basic engineering and design principles. Such extensions to our robotics work should continue to enhance the quality of our multidisciplinary instructional efforts.

Bibliography

1. Beer, R., Hillel, C., and Drushel, R., "Using Autonomous Robotics to Teach Science and Engineering," *Communications of the ACM*, Vol. 42, No. 6, June 1999, pp. 85-92.
2. Martin, F., Mikhak, B., Resnick, M., Silverman, B., and Berg, R., "To Mindstorms and Beyond: Evolution of a Construction Kit for Magical Machines," *Robots for Kids: Exploring New Technologies for Learning*; A. Druin and J. Hendler, eds., Morgan Kaufmann, 2000, pp. 9-33.
3. Saffo, P., "The Consumer Spectrum," in *Bringing Design to Software*, T. Winograd, ed., Addison-Wesley, 1996.
4. Knudsen, J., *The Unofficial Guide to LEGO Mindstorms Robots*, O'Reilly & Associates, 1999.
5. Turkle, S. and Papert, S., "Epistemological Pluralism and the Revaluation of the Concrete," *Journal of Mathematical Behavior*, Vol. 11, No. 1, March 1992, pp. 3-33.

6. Miller, G., Church, R., and Trexler, M., "Teaching Diverse Learners Using Robotics," *Robots for Kids: Exploring New Technologies for Learning*, A. Druin and J. Hendler, eds., Morgan Kaufmann, 2000, pp. 165-192.
7. Norstrand, B., "Autonomous Robotics Projects for Learning Software Engineering," *Proceedings of the 2000 IEEE International Conference on Systems, Man & Cybernetics*, Nashville, August 2000, pp. 724-729.
8. Avanzato, R., "Mobile Robotics for Freshman Design, Research, and High School Outreach," *Proceedings of the 2000 IEEE International Conference on Systems, Man & Cybernetics*, Nashville, August 2000, pp. 736-739.
9. Jadud, M., "TeamStorms as a Theory of Instruction," *Proceedings of the 2000 IEEE International Conference on Systems, Man & Cybernetics*, Nashville, August 2000, pp. 712-717.
10. Meeden, L., "Using Robots as Introduction to Computer Science," *Proceedings of the Ninth Florida Artificial Intelligence Research Symposium*, Key West, 1996, pp. 473-477.
11. Gaines, D., and Balac, N., "Using Mobile Robots to Teach Artificial Intelligence Research Skills," *Proceedings 2000 ASEE Annual Conference*, St. Louis, June 2000.
12. Kumar, D. and Meeden, L., "A Robot Laboratory for Teaching Artificial Intelligence," *Proceedings of the Twenty-Ninth SIGCSE Technical Symposium on Computer Science Education*, Atlanta, 1998, pp. 341-344.
13. Russell, S. and Norvig, P., *Artificial Intelligence: A Modern Approach*, Prentice Hall, 1994.
14. Arkin, R., *Behavior-Based Robotics*; The MIT Press; 1998.
15. Brooks, R., "A Robust Layered Control System for a Mobile Robot," *IEEE Journal of Robotics and Automation*, Vol. RA-2, 1986, pp. 14-23.

JERRY B. WEINBERG
Jerry B. Weinberg is an Assistant Professor in the Computer Science Department at Southern Illinois University - Edwardsville. He teaches courses and conducts research in artificial intelligence and human-computer interaction. Dr. Weinberg received a B.S. degree in Nursing from Indiana State University (1984), a B.S. degree in Computer Science from The University of South Carolina (1988), and his Ph.D. in Computer Science from Vanderbilt University (1996).

GEORGE L. ENGEL
George L. Engel is a Professor of the Electrical and Computer Engineering Department at Southern Illinois University - Edwardsville. His interests include electronics, VLSI design, computer system design, automated design, and fabrication tools. Dr. Engel received his D.Sc. in Electrical Engineering from Washington University in 1990.

KEGIN GU
Keqin Gu is an Associate Professor and the Graduate Program Director of Mechanical Engineering, Southern Illinois University - Edwardsville. He teaches and conducts research in the general areas of dynamic systems, control and robotics. He is an Associate Editor of *IEEE Transactions on Automatic Control*. Dr. Gu received his Ph.D. in Mechanical Engineering from the Georgia Institute of Technology in 1988.

S. CEM KARACAL
S. Cem Karacal is an Associate Professor of Industrial Engineering at Southern Illinois University - Edwardsville. His main research and teaching interest areas are simulation modeling, quality control, operations research, and facilities layout. Before joining SIUE, he worked at the Rochester Institute of Technology as a faculty member and served as the Computer Integrated Manufacturing System project coordinator for RIT's integrated circuit factory. Dr. Karacal received his Ph.D. in Industrial Engineering from Oklahoma State University in 1991.

SCOTT R. SMITH
Scott R. Smith is a Professor in the Electrical and Computer Engineering Department at Southern Illinois University - Edwardsville. His interests include computer architecture and design, multimedia applications, biomedical engineering, computer based education systems, and consumer electronics. Dr. Smith received his Ph.D. in Electrical Engineering from the University of Illinois at Urbana-Champaign in 1991.

WILLIAM W. WHITE
William W. White is an Associate Professor in Computer Science at Southern Illinois University - Edwardsville. He received his Ph.D. at Ohio State University in 1989 and has performed R&D in his specialties of computer graphics and networking with positions at IBM, Los Alamos National Laboratory, Argonne National Laboratory, Walt Disney Feature Animation, and the University of North Dakota.

XUDONG WILLIAM YU
Xudong William Yu is an Assistant Professor in the Computer Science Department at Southern Illinois University - Edwardsville. He received his Ph.D. in Computer Science from Vanderbilt University in 1992. His research interests include model-based reasoning and diagnosis, hybrid systems for diagnosis, knowledge-based systems, modeling and analysis of complex systems, and machine learning.