ENHANCING UNDERGRADUATE LEARNING THROUGH MOBILE ROBOTICS

Carlotta A. Berry, Ph.D.
Department of Electrical and Computer Engineering
Rose-Hulman Institute of Technology

Abstract

This paper presents the development and implementation of an introductory mobile robotics course at Rose-Hulman Institute of Technology (RHIT). This course is one of the last courses in the multidisciplinary robotics minor available to computer science, computer, electrical, mechanical and software engineering students. The objective of this course is to present robotics applications and theory while also providing the students with an appreciation of their discipline and how it applies to other disciplines. The diversity of students in this course means that they will experience a more realistic model of their future workplace dynamic and demographic. The results of the first three offerings of this course will be presented as well as the lessons learned and recommendations for improvement.

Introduction

Undergraduate students in science and engineering frequently express a desire to relate the abstract theory presented in class to real-world or practical application. One method that can be used to integrate component theory with system or practical application is robotics. Since robotics theory includes topics such as sensors, controls, mechatronics, kinematics, microcontroller programming, embedded systems and software development; it is an ideal model for multidisciplinary application. Students from several disciplines including electrical and computer engineering, computer science and software engineering and mechanical engineering can work together in a robotics course to gain depth understanding of their major and breadth understanding of another major. It is hypothesized that this type of classroom experience is a more realistic simulation of their future workplace.

Robotics is typically used as an artifact to engage K-12 students in science, technology, engineering and mathematics (STEM), recruit students to STEM, teach concepts such as programming, controls or embedded systems and also to teach freshman or senior design. Based upon mainstream media, movies, film and K-12 competitions; students gain interest in robotics but also at times an unrealistic perception of the state of the art. Typically, undergraduate robotics courses suffer from lack of a good textbook and either too basic or too complicated activities because of student pre-requisite knowledge and skill.

This paper will present the details of the development and offering of an upper level course, "*Introduction to Mobile Robotics (IMR)*", designed to teach multidisciplinary robot theory and application that also gives the students an appreciation for some of the open research issues and challenges. The IMR course is one of the last courses in the multi disciplinary robotics certificate program at RHIT, so some students have prior experience with robotics (http://robotics.rose-hulman.edu). This course is innovative in the fact that it is available to students from multiple disciplines and attempts to motivate students for further study or research in robotics versus using the robot as a tool to motivate some other topic. This is compelling because these students are already interested in STEM fields and have or will take courses in design, software, programming, electronics, controls and kinematics. Since this course is taught at a primarily undergraduate engineering institution it is of particular importance that the course also serves as a recruiting tool for undergraduate or graduate research as well.

This course will provide the student with a synthesis and evaluation of engineering and science concepts learned in prior courses. It will not only include mobile robot theory but the implementation of behaviors and control algorithms on an actual mobile robot. Students will gain exposure to the theory but also some of the challenges such as sensor and odometry error and bandwidth limitations. Finally, students will learn about a topic that interests them, engages them in multidisciplinary work, corrects some common robotics misconceptions and potentially recruits students for research.

Literature Review

The goals of this special topics course in mobile robotics are to teach students about robotics history, theory and research while they also gain an appreciation for multidisciplinary work. One of the first steps in the design of the course was to review the literature and identify other courses with similar goals and objectives. This search produced many robotics related courses but surprisingly few with the objective of teaching robotics. In other words, it was difficult to find courses that were using robots to teach robotics (i.e. robots for robotics sake). The diversity of courses found provided more evidence that it is indeed an ideal multidisciplinary tool for teaching concepts in science and engineering. Some of the courses were at the pre-college level and robots were used to increase or maintain students' interest in science and mathematics [1-5]. These courses typically used LEGOS, RugWarrior and a Handy Board microcontroller. Some of these courses and activities were to prepare students for competitions such as FIRST and Botball, which have proven successful for recruiting students to science and engineering.

At the collegiate level, there were courses for underclassmen to introduce them to programming, computer science and engineering concepts [6–17]. The vast majority of these courses were in computer science, electrical, computer and mechanical engineering departments. These courses were overwhelmingly single discipline with only a few cross listed in multiple departments. There were also several courses that used robots to teach microprocessors, microcontrollers, and embedded systems concepts [18 – 23]. Robotics has also been used to provide students with a multidisciplinary team experience as they learn the engineering design process [24- 41]. In most of these courses, the students would design and build a LEGO robot to accomplish a given task. The controller for these courses was typically the Basic Stamp or Handy Board controller. Furthermore, some of the authors even surmised that robot design can be used to satisfy ABET core outcomes a – k as well [29-31]. Table 1 presents a summary of related courses that had components similar to the mission of the IMR course. It should be noted that many of these courses were discipline-specific and may have used the robot to motivate another topic as well.

Table 1: Summary of Related Courses.

School	Content Summary	Hardware
Brown University	Embodied Gaming	Roomba [42]
Drexel University	CS, AI, engineering problems	LEGOS w/HandyBoard [43]
University of West Florida	Curriculum integration	LEGOS w/HandyBoard [44]
Swarthmore College	Research project preparation, AAAI	Khephera, ActivMedia Pioneers [45]
Missouri University of Science and	State of the art of robotics and architectures	Instructor-created kit using embedded C, Matlab

Technology		image processing [46]
Carnegie Mellon University	Robots for study problem-based laboratory experiments	LEGOS with HandyBoard [47-48]
Pontificia Universidad Catolica de Chile	mobile robot programming for autonomous navigation	ER1 Mobile Robot [49]
Augsburg College	CS course on robot history and theory	Robix Manipulator, instructor-created vehicle [50]

Course Format

The first offering of the IMR course was in spring 2007 and it quickly became apparent that the proposed topics were too ambitious. The topics included simulation, actuators, effectors, locomotion, kinematics, sensors, control, navigation, localization, path planning, computer vision, image processing, human-robot interaction and GUI design. The challenge was that some of these topics were entire courses in themselves (i.e. computer vision, human-robot interaction). Although it was preferable to prepare students for robotics research after one quarter, it was soon discovered that it was just not feasible. The author concluded that it was more appropriate to focus on system level artificial intelligence (AI) techniques and assume that prior courses addressed some of the most basic components of a robot. The second offering of the course was in spring 2009; the list of topics was greatly pared down and the result was that the course was more effective. If this course continues to be successful then the eliminated topics will be included in a subsequent course or alternate course (i.e. advanced mobile robotics).

The grading scale was also changed to put more weight on the final project, laboratory assignments and daily reading quizzes. This was because with the amount of programming required to implement AI techniques on the robot, it was not possible to also assign a significant amount of homework or exams. This change was to encourage the students to do the required reading and review the concepts presented in class daily. Robotics is a topic that requires a continuous focus versus intermittent review. This model did appear to work better for getting the students engaged in the material and not just the robot. Furthermore, there were less late submissions of the laboratory assignments and the quiz grades were relatively high. One additional change for the next offering was that the quizzes were closed book and notes with a stricter time limit. Using the original model, the quizzes were designed to be completed in 5 to 10 minutes and it sometimes took the students in excess of 30 minutes because they searched for answers in the textbook. It was also observed during the first course that it was necessary to correct student misconceptions that the course was all about “playing” with a robot. The use of the quizzes insured that the students learned something about the subject matter as well by requiring them to answer basic robotics history and theory questions.

Originally, before the robotics certificate curriculum, this course had no prerequisites other than junior level classification and programming proficiency. It was open to any major who met these requirements. It was soon discovered that students overestimate their programming and robotics ability and having lax prerequisite requirements allowed students to enroll who were not prepared for the level of rigor of this course. Thus, in the subsequent quarters, the prerequisites were changed to control systems and programming proficiency or instructor permission. This change served as a filter for the course to allow upper level students and those who were serious about the subject matter to enroll. The students’ major was not a consideration as much as their ability to meet those requirements. It was believed that the student could learn any of the basic electronics or mechanics, if the desire was there.

For example, since Computer Science students do not typically take controls, supplemental instruction or independent study on this subject matter would be required, if necessary. Instructor permission to waive the prerequisite requirements was reserved for students enrolled in the robotics certificate curriculum or who have prior exposure with robotics (i.e. FIRST, BotBall). In this way, the success of this course and the students' success in this course were not so closely tied to their unrealistic expectations of what they could do with a robot and what a robot could do. This was the lesson learned after the first offering of the course when the students really struggled to program the robot and complete the labs.

Originally the course was offered 4 days a week with 3 days of one-hour lecture and one 3-hour lab period. The lab session was for last minute code revisions and robot demonstration. After the first offering, it was determined the students needed more class time with the robot. Even though the students were allowed to check the robot out and take it home, they were rarely able to meet the lab assignment submission deadlines. This shortfall could be attributed to two factors: instructor and student inexperience. Since this was the first offering of the course, the instructor overestimated the students' programming abilities and the lab expectations were too difficult. Secondly, the students overestimated their abilities and did not log the required 8 hours per week working with the robot outside of class. Therefore, in the second offering of the course, the format was changed to 3 days per week, two hours per day. The first two days included one hour of lecture and one hour of lab recitation or lab work. The last day of the week was for lab completion and demonstration. This allowed the students to work with the robot for at least an hour every day. This change gave the students more opportunities to ask questions and gauge whether their progress was reasonable by observing their peers. This resulted in more of the students completing the laboratory assignments in a timely manner. However, it did reduce the amount of lecture time and put more responsibility on the students for independent study and reading. Despite this, it appeared that the students were able to obtain a more depth understanding of the required robot theory and application and a greater since of accomplishment with the robot. The last two weeks of the course did not include lecture because the students used that time to prepare for the final project and demonstrate milestones.

Lectures

This course was taught for the third time in spring 2010. The textbook, lectures and labs were changed again in order to continuously improve the content and converge on the overall course objective. Since the textbook is the template for the course, it is important to select one that is appropriately detailed with relevant coverage. However, it was very difficult to find such a textbook for an undergraduate multidisciplinary mobile robotics course that is not too advanced or too basic for the objectives of the course. Some textbooks overly simplified key robotics concepts and focused primarily on depth coverage of hardware, typically LEGOs (i.e. K-12 level). Other textbooks presented depth coverage of higher-level concepts in controls and probability and neglected the presentation of basic applications (i.e. graduate-level). Still, there were other textbooks that were very fundamental or introductory and did not provide enough detail to implement the techniques on an actual hardware platform (i.e. underclassmen-level). This divergence in presentation created great difficulty in the ability of the student's to master the material and apply it to the physical robot. One solution to this problem was to provide more in-class lectures regarding the implementation on the mobile robot. Similar to what other courses have done, the IMR course is slowly moving toward no formal textbook in lieu of handouts, research papers, web resources or a course packet. In order to present the higher level topics in AI robotics, some of the lower level topics such as effectors, actuators, locomotion, and sensors were sacrificed or introduced in a just in time lecture. However, it was still necessary to keep some lectures on kinematics and classical control such as PID. It is believed that this modification prepared the student to complete the laboratory assignments and final project in a more effective and efficient manner. Based upon several student requests, some brief lectures on the Visual C# IDE and programming concepts such as threading were added during the first weeks' lecture. The lecture format was typically a multimedia presentation with PowerPoint, images, videos and some active learning activities such as partial lecture notes, collaborative think-pair-share and paired programming or coding.

Hardware

One of the biggest dilemmas in the design of this course was the selection of the robot platform. As previously mentioned, although the LEGO Mindstorm was a very popular choice, the author felt that the students needed a platform that was less simplistic and capable of a more diverse sensory suite. Since many students have been exposed to LEGOS in K-12 or other undergraduate courses, the reasoning was that they may not view this platform at the desired higher level required to treat it as a tool for traditional robotics research and application. This platform actually occludes some of the hardware and programming dilemmas that the student should experience in order to appreciate the state of the art. For similar reasons, the Creates (Roombas) programmed using Python, which are used in the introductory programming and software development course at RHIT were not selected.

In 2007, the hardware platform was a Traxster I robot programmed using a Microchip PIC 18 microcontroller. The students were able to implement wall following, obstacle avoidance, follow center, and follow robot behaviors on the mobile robot. In order to implement these behaviors, the students integrated concepts from courses in controls, mechatronics, communications, microcontrollers, and programming. However, the students experienced many problems writing the C code to set up the most fundamental hardware components such as driving the motors. Therefore, it made it very difficult from them to quickly implement high level behaviors. Thus, in 2009, the controller on the robot was changed to a Robotics Connection Serializer. The Serializer robot controller is an interface to the Microsoft.NET framework and allows an easy interface to DC motors, servos, analog sensors, I2C slave devices, single and quadrature encoders, and switches/relays. The benefit of using the Serializer is that it was possible for the students to implement artificial intelligence algorithms on the robot quicker because it occludes all of the low-level programming such as setting bits to interface to hardware. Students were able to use the Serializer libraries in order to quickly send commands and receive data from the robot hardware. The Serializer has dual 4A H-bridges to control 2 DC motors, built in velocity and distance PID control algorithms, I2C port, 6 10-bit analog inputs, 13 I/O lines (6 for servo control), and dual encoder ports. The goal of this controller is to get the robot up and running quickly without the distraction of setting bits.

The robot was also upgraded to a Traxster II with better motors with quadrature encoders in order to improve odometry error. The tracks were removed from the differential drive Traxster and replaced with LEGO kit wheels and a caster due to numerous problems with slippage, the skid turn and severe inaccuracies in odometry. The new robot also included a greater diversity of sensors which afforded flexibility in the laboratory assignments and the potential solutions. This change also helped because the addition of sensor redundancy gave solutions options for combating the ever present sensor error. Table 2 presents the robot sensors and peripherals.

Table 2: Robot sensors and peripherals.

Buzzer	Keypad	Sonar Sensor
Camera	LCD Display	Compass
Thermopile Array	Infrared Sensor	Text to Speech Synthesizer
Line Following Sensor	Pushbutton	
	I/O Board	

Software

Originally, the PIC18 microcontroller was used for the course and it was programmed in PICCLITE using MPLAB. Although the students were able to implement basic robot behaviors, some of them expressed a desire to use Microsoft Robotics Studio (MSRS) to quickly implement more high-level intelligence. As previously mentioned, the reason for the desire to do more was because programming at the bit level could be cumbersome and it took the students a long time to set hardware configurations, timers, interrupts, etc. There was a significant amount of code and hardware preparation just to get the robot moving. Therefore in 2009, the controller was changed to the Robotics Connection Serializer that could be programmed with Visual C# using MSRS services [54-55]. For example, some of the more basic robot peripherals such as IR, sonar, speech and PID motor controller were already classes in the library and the methods and functions could be called immediately. This format was actually more appropriate because it moved the focus from components to higher level functions and behaviors. This change required students to view the robot from a systems-level perspective that had to globally interact in order to exhibit a certain level of intelligence. The object oriented programming in the Microsoft Visual Studio IDE also afforded the creation of a GUI to make it possible to more easily visualize the framework of the student's work. By changing to this software platform, students were able to accomplish more basic robot behaviors quicker because they were not bogged down in setting bits as opposed to using a simple function call. These functions were immediately available to control actuators and poll sensors.

It should be noted that in all of the versions of the course, students were given starter code but even with this assistance there was a significant difference in their performance between the course before and after the introduction of the Serializer with MSDN libraries. Interestingly, even in the new format, some students did not want to use the PID motor controller available in the Serializer library but rather desired access to the hardware such as encoder counts, event and timer interrupts in order to create their own motor controllers and sensor functions. It was important to emphasize to the students that there are always tradeoffs in the selection of any hardware and software platform but despite these, it was important to create the best possible solution with what was available. The author feels that using the higher level language was more beneficial for the study of robotics theory and more appropriate for the goals of this course and that some of these additional capabilities actually diverge from the purpose. Not only was it not an objective of the course for the students to implement a PID motor controller from scratch, it was actually counterproductive. In fact, these types of concepts and activities would be covered in a mechatronics or controls course. At times, the desire to work at that level was a struggle between the instructor and the students but it did present a great teaching moment. In their future career, they may not always have the luxury of selecting their tools, hardware, software or the overall system but rather they will be presented with a problem and they must devise the best problem solution even if the environment is not ideal. This may mean that they have to use a controller where some of the features or functions are a "black box" but they still must successfully complete the desired task.

Labs

The inspiration for the laboratory assignments was the course topics, other robotics courses, and three textbooks [51-53]. The purpose of the laboratory assignments was to expose the students to robot applications founded in the essential theory. This included the implementation of basic robot behaviors such as wall following, obstacle avoidance and navigation to achieve prescribed tasks. During these assignments, the students also encountered some challenges in robotics research such as odometry error, sensor noise and bandwidth limitations. Although, the students may not have always been able to resolve these issues, it is hoped that the experience caused them to think about the field of robotics from a more realistic perspective and possible resolutions.

Final Project

The final project for spring 2007 was a competition similar to a relay race. This project combined several of the robot behaviors implemented during the quarter. As part of the competition, students used wall following, follow center of the hallway, object following and obstacle avoidance to move the robot to a goal point. The students overall score was based upon time and the robot completing each task. Bonus points were awarded for the high scorers. The final project for spring 2009 was a navigation task where students used metric path planning to move the robot from a start to a goal point for several worlds. The students score was based upon accuracy and time and the high scorers received bonus points. In 2010, this task was further complicated with the integration of localization in order to rescue a kidnapped robot and drive it home. More details regarding the final project are provided in the results section of the paper and on the course website.

Results

This section will present the results of the first offerings of the IMR course. Due to hardware limitations, the enrollment in the course was limited to 18 and 14 students, respectively. It should be noted that a typical class size at Rose-Hulman is 20 to 30 students. The reason for the strict limitation on the enrollment was based upon the lesson learned from the first offering. Ideally, there should be enough spare robots for 50% of the class. Unfortunately, there is no simulator available for the Traxster robot so by using this rule, the students continue to make progress on their laboratory assignments while their primary robot was repaired. Typically, the hardware failures were with Bluetooth modules, Serializer boards, track links, and wiring problems. Since there are 10 robots available for the course, the enrollment was limited to 14 students separated into teams of two. Figure 1 provides the course demographics for the two offerings of the IMR course.

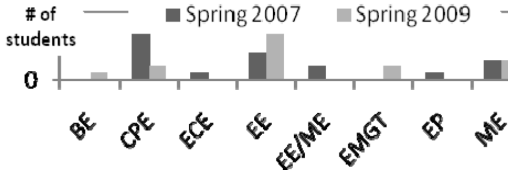
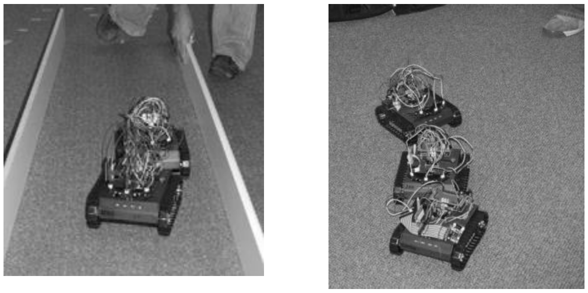


Figure 1: Course Demographics.

It should be noted that since the robotics certificate program is only in its second year, it is believed that the diversity of majors, classifications will continue to improve. Typically, 90% of the students are seniors with 10% being juniors or graduate students. Currently, 10 students have graduated from the robotics certificate program and they completed the IMR course in 2009 and 2010. The robotics certificate faculty are also researching the possibility of opening up the robotics minor to biomedical engineering students which would also significantly increase the number of female students who enroll in the course.

Labs

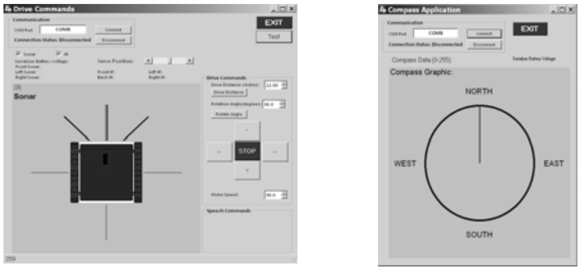
In spring 2007, one of the more successful lab experiments was the implementation of follow center, follow object and follow robot behaviors on the Traxster I. The robot had 4 infrared sensors mounted on the chassis and 3 mounted on the servo. Reactive control was used to program the robots to follow a given trajectory until it encountered objects on both sides (i.e. a hallway). The robot would then adjust its trajectory to drive forward down the center of the hallway. For the follow object or follow robot behavior, the robot attempted to follow an object in the front while maintaining a distance of 5 inches. While following the object, if another object appeared closer, the robot abandoned the first object and attempt to follow the new one. The students were required to create the pseudo code, flowchart and then demonstrate the final design on the physical robot. Figure 2 demonstrates the Follow Center and Follow Robot behaviors. As part of each week's lab report, the students were required to reflect on the essential theory, challenges encountered, how to address these challenges and how to improve the robot's behavior and/or laboratory assignment.



a. Follow Center b. Follow Robot

Figure 2: Sample Robot Behaviors.

In spring 2009, the students were provided with starter Visual C# code for motor and servo control and polling sensor data from the Serializer including the IR sensors, sonar, thermopile array, compass, line following sensor and pushbuttons. The starter code was provided in the form of a GUI with the underlying code. Figure 3 presents a sample of two of the GUIS that the students were given.



a. IR, Sonar, Drive App b. Compass App

Figure 3: Visual C# GUI Screen shots.

In one of the first labs of the quarter, the students implemented a wall following behavior on the robot using open loop control. It should be noted that this was one of the last labs when the course used the PIC18 microcontroller. In the subsequent lab, the wall following algorithm was improved by using feedback control. A proportional-derivative controller was used to move the robot along a wall for at least 4 feet while maintaining a distance from the wall of 4 to 6 inches. The robot negotiated obstacles, corners and doorways with minimal contact while continuing to follow the wall. Figure 4 provides a graphical illustration of the robot's behavior in 2 different environments.

a. artificial environment b. real world

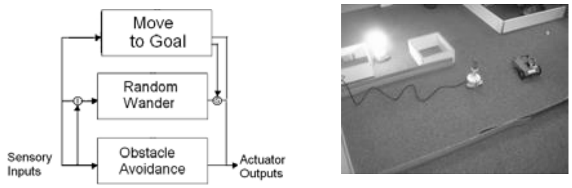
Figure 4: Wall Following Example.

The students were to consider how this new controller affected the robot's performance with respect to overshoot, transient and steady state errors. Some student teams were able to program the robot to maintain contact with the wall around corners, out of the doorway of the classroom and down the hall in the building for at least 12 yards.

In this same lab, some students chose to improve the line following algorithm created in a prior lab by incorporating a proportional-integral controller. Due to lighting inconsistencies, odometry issues such as robot overcorrection and bandwidth limitations when polling the line sensor; this assignment was not quite as successful as the wall following. There were severe oscillations and many instances of overshooting the line. If the robot started on the line, moved slowly in order to reduce sensor aliasing and used a finite state machine to keep track of how many of the individual sensors were activated, it performed better than if it started off of the line or overshot the line and had to use a smart wander routine to find the line to follow. Most students were not able to accomplish this task on any level until the original path was greatly simplified. The students were to consider how this new controller affected the robot's performance with respect to overshoot, transient and steady state errors. Also, the students were to address the speed of circumventing the path and the ability to find the line when lost based upon the controller design.

The homing or docking lab was implemented on the mobile robot by using hybrid control. A heat beacon was placed in the robot's environment and the goal of the lab was for the robot to use a priori information about the environment to plan a path to the beacon and come within one foot of it without hitting it. The partial world map (representation) included metric distance and direction to the beacon with respect to the robot's current pose. This representation was the input to the deliberative layer of the architecture. Updates to the path were based upon sensor feedback from the distance, heading and thermopile sensors. The middle layer was used to make decisions about whether path updates were handled in the deliberative or reactive layer. The reactive layer handled obstacle avoidance. Once the robot was close enough to sense the beacon with the temperature sensor, it used this directional information to continue toward it. During this lab the students were to consider dilemmas such as what happens when there are dynamic changes to the environment while the robot executes a plan. How well did the robot respond to different starting positions and beacon locations? How could a more detailed world map improve the homing algorithm? How did to handle the compass sensor inconsistencies in the design of the homing routine?

Finally, the homing and docking lab was improved by implementing a reactive (behavior-based) control. The robot used either random wander and obstacle avoidance or a smart wander or cover behavior to move in the environment until the heat beacon was sensed. The robot would then execute a move to goal behavior based upon the information from the thermopile array. This algorithm was based upon the subsumption architecture where the obstacle avoidance was the lowest level and received the highest priority. During this lab, students were to consider how the robot's performance compared to the hybrid control. Did it find and move to the beacon quicker? Was there a real benefit in having a world model for the robot? Figure 5 presents the control architecture and images from the hybrid control lab.



control architecture

Heat Seeking Robots (Light Bulb and Heater)

Figure 5: Homing and Docking Laboratory Assignment Images.

Final Project

In spring 2009, the final project was mapping and navigation but changed to just navigation when the mapping component proved to be too difficult. The metric path planning and execution portion of the project involved using a wavefront algorithm to create a path from the robot's start position to goal location. The robot's obstacle avoidance and move to goal behaviors were used to move through the list of goals points until the robot arrived at the final destination. The algorithm used an eight-neighborhood so that the robot could move diagonally however; a four-neighborhood would have also worked. The test arena was 6 ft x 6 ft with 1 ft x 1 ft obstacles. The configuration space was an occupancy grid divided into 6" x 6" squares, where free space was denoted by '0's and occupied space by '99's. The students designed a scheme to represent the robot's start position and goal location such that these values were specified at run time. During the demonstration, the students were given the world map, generated the wavefront and planned the path from the start to goal and the robot then executed the plan. The students were graded on the ability of the robot to reach the goal while avoiding obstacles and the efficiency of the path chosen or time. Some of the strategies that the students used to accomplish the navigation task were to grow the obstacles to avoid collisions and to completely remove any spaces that the robot could not fit through from the given map in order to speed up the algorithm. Many of the students used a GUI to display the metric or topological map and all of the robot's path options. One student group actually derived an algorithm to select the path based upon minimizing steps, turns, or distance that could be selected from the GUI. Students received bonus points if they were able to use sonar and infrared sensors to create a map of the artificial environment. One student team was not only able to use the robot to make a partial world map but to use the wavefront algorithm to plan a path from a start to goal location on this map. In spring 2010, localization was added to the final project and it was successfully implemented by all of the student teams. The algorithm used a Partially Observable Markov Decision Process to identify landmarks and gateways on the map representation and then using sensor feedback from the robot it would move between landmarks until it could localize with 100% certainty. Upon successful localization, the robot was then required to plan a path to the home position by using grassfire (wavefront) expansion.

Conclusions and Future Work

This paper has presented the details of the implementation of an IMR course by reviewing the related literature, providing the course details and the results of the first three offerings. It is evident that designing a course to teach the history, theory and application of robotics has been a windy road. However after multiple offerings, the author is confident that the course is converging on the proper balance of theory and application. The students and instructor are slowly becoming more proficient at achieving the course goals. It is believed that students are not only gaining an appreciation for the state of the art but also having fun. They developed a realistic perspective of the mobile robot's capabilities, open areas of research and the importance of multidisciplinary teamwork. Lastly, ten robotics certificate students completed the course and graduated in 2009 and 2010. Three of them went on to careers in controls, robotics and automation and two went on to graduate study in robotics. The feedback from two of these students indicated that this course was helpful and relevant to their current positions. One graduate student indicated that the concepts learned in the IMR course have proven helpful in his research program. The student working in automation indicated that the format used for this course modeled the closest to his actual workplace environment. In addition, there are 15 students on track to obtain the robotics certificate (minor) in spring 2011.

Despite the many successes of the IMR course, there is always room for improvement. Some of the planned future work involves changing the lectures and assignments to include more research and AI theory. Research papers will be integrated into the required reading and possibly student presentations on the readings. Quizzes will be changed to closed book, closed notes and limited to ten minutes. Labs will continue to transition to higher level AI tasks. The final project will eventually become a mapping task with localization or SLAM (simultaneous localization and mapping) and navigation. The robot controller will be changed or improved and options explored to integrate the CMU camera to work with the current Serializer controller. After this change, the CMU camera will be used for a vision-based lab integrated with robot behavior and motion versus a stand-alone lab. The change to the controller would help bandwidth limitations based upon the Bluetooth radio. Possibilities include changing to XBee communication on the same controller. Another option would be to change to an Arduino controller which could be programmed tethered to the students' laptop and but run as a standalone program that communicates robot status and data to the laptop. Additionally, due to the electrical interference in the building and severe inaccuracies, the compass was removed from future labs. Finally, students will be provided with one GUI to control and poll all of the robot's peripherals on the first day of class. This GUI will demonstrate all of the robot's capabilities in one compact form versus giving the students weekly code snippets. The reason for this change is that the students will have more flexibility in design decisions and the preferred method to accomplish the laboratory assignment requirements. This may also enable the laboratory assignments to be more open ended. There is more information about the IMR course at the course website: (http://www.rose-hulman.edu/~berry123/Courses/ECE497.html).

Acknowledgement

The author would like to thank the Rose-Hulman Electrical and Computer Engineering department and Rockwell Collins for providing financial support to purchase the robot and peripherals for this course.

Bibliography

1. B. Brand, M. Collver, and M. Kasarda, "Motivating Students with Robotics", *The Science Teacher,* April/May 2008, pp. 44 – 49.

2. S. Bruder and K. Wedeward, "Robotics in the Classroom", *IEEE Robotics and Automation Magazine,* September 2003, pp. 25 – 29.

3. E. Matson and S. DeLoach, "Using Robots to Increase Interest of Technical Disciplines in Rural and Underserved Schools", in *Proceedings of the 2004 ASEE Conference and Exposition*, Salt Lake City, UT, June 20-23, 2004.

4. I.R. Nourbakhsh, K. Crowley, K. Wilkinson, and E. Hamner, "The educational impact of the robotic autonomy mobile robotics course", technical report, CMU-RI-TR-03-29, Robotics Institute, Carnegie Mellon University, August 2003.

5. M. Petre and B. Price, "Using robotics to motivate 'back door' learning", *Education and information technologies,* Volume 9, Number 2, pp. 147 – 158, 2004.

6. D. Chang, P. Hanlon, K. Ingold, and R. Rabb, "Educating Generation Y in Robotics", in *Proceedings of the 2009 American Society of Engineering Education Annual Conference and Exposition,* Austin, TX, June 14 – 17, 2009.

7. D.C. Cliburn, "Experiences with the LEGO Mindstorms throughout the undergraduate computer science curriculum", in *Proceedings of the 36th ASEE/IEEE Frontiers in Education Conference,* San Diego, CA, October 28 – 31, 2006.

8. D.L. Duke, J. Carlson, and C. Thorpe, "Robotics in Early Undergraduate Education", in *Proceedings of the 2007 American Association for Artificial Intelligence (AAAI) Symposium on Robot and Robot Venues: Resources for AI Education*, Palo Alto, CA, March 2007.

9. M. Gini, "Learning Computer Science through Robotics" in *Proceedings of the 1996 American Society of Engineering Education Annual Conference and Exposition*, Washington, DC, June 23 – 26, 1996.

10. F. Klassner and S.D. Anderson, "Lego Mindstorms: Not just for K-12 anymore", *IEEE Robotics and Automation Magazine,* June 2003, pp. 12 – 18.

11. D.J. Livingston, and J.C. Squire, "Robotics in introduction to electrical and computer engineering at the Virginia Military Institute", in *Proceedings of the 2002 American Society for Engineering Education Annual Conference and Exposition,* Montreal, Canada, June 16 – 19, 2002.

12. A. Mentis, C. Reynolds, D. Abbott-McCune, and B. Ring, “Cementing abstraction with a concrete application: a focused use of robotics in CS1”, in *Proceedings of the 2009 American Society of Engineering Education Annual Conference and Exposition,* Austin, TX, June 14 – 17, 2009.

13. D.J. Pack, and A.R. Klayton, “Education through robotics at the United States Air Force Academy”, in *Proceedings of the World Automation Congress (WAC),* Budapest, Hungary, July 24 – 26, 2006.

14. G.L. Plett and M.D. Ciletti, “Piloting a balanced curriculum in electrical engineering – Introduction to robotics”, in *Proceedings of the 2005 American Society of Engineering Education Annual Conference and Exposition,* Portland, OR, June 12 – 15, 2005.

15. C. Polalaza-Raez and B.H. Groff, “Retention 101: Where robots go…students follow”, *Journal of Engineering Education,* January 2003, pp. 85 – 90.

16. W. White, J.B. Weinberg, G.L. Engel, S.C. Karacal, S.C., and A. Hu, “Assessing an interdisciplinary robotics course”, in *Proceedings of the 2005 American Society of Engineering Education Annual Conference and Exposition,* Portland, OR, June 12 – 15, 2005.

17. J.B. Weinberg, and X. Yu, “Robotics in Education: Low cost platforms for teaching integrated systems”, *IEEE Robotics and Automation,* June 2003, Vol. 10, No. 2, pp. 4-6.

18. B. Avanzato, “A robotics-based microprocessor course for engineering technology”, in *Proceeding of the 1996 American Society of Engineering Education Annual Conference and Exposition,* Washington, DC, June 23 – 26, 1996.

19. S. Barrett, J. Anders, J. Hamann, R. Kubichek, R., and S. Muknahallipatna, “Embedded System Design: Responding to the Challenge”, in *Proceedings of the 2009 American Society of Engineering Education Annual Conference and Exposition,* Austin, TX, June 14 – 17, 2009.

20. S. Barrett, D.J. Pack, P. Beavis, M. Sardar, A. Griffith, M. Stephens, J. Sandberg, L. Sircin, and G. Janack, “Using Robots to Teach Complex Real Time Embedded Systems Concepts”, in *Proceedings of the 2005 American Society of Engineering Education Annual Conference and Exposition,* Portland, OR, June 12 – 15, 2005.

21. M.L. Neilsen, D.H. Lenhert, M. Mizuno, G. Singh, J. Staver, N. Zhang, K. Krmaer, W.J. Rust, Q. Stoll, and M.S. Uddin, “Encouraging interest in engineering through embedded system design”, in *Proceedings of the 2004 American Society of Engineering Education Annual Conference and Exposition,* Salt Lake City, UT, June 20 – 23, 2004.

22. B. Surgenor, K. Firth, and P. Wild, “Lessons learned from a mobile robot based mechatronics course”, in *Proceedings of the 2005 American Society of Engineering Education Annual Conference and Exposition,* Portland, OR, June 12 – 15, 2005.

23. W. Walter, “A micro-controller based robotics course for ME students”, in *Proceedings of the 2002 American Society of Engineering Education Annual Conference and Exposition,* Montreal, Canada, June 16 – 19, 2002.

24. D.K. Blandford, D.J. Hwang, and A. Richardson, “LEGO 101: A multidisciplinary freshman team experience”, in *Proceedings of the 2001 American Society for Engineering Education Annual Conference and Exposition,* Albuquerque, NM, June 24 – 27, 2001.

25. J.E. DeVault, “Robot Stories: Interdisciplinary Design with Autonomous Mobile Robots”, *Computers in Education Journal*, Volume 10, No. 1, 2000, pp. 21 - 27.

26. L.B. Huettel, A.S. Brown, K.D. Coonley, M.R. Gustafson, J. Kim, G.A. Ybarra, and L.M. Collins, “Fundamentals of ECE: A rigorous, integrated introduction to electrical

and computer engineering", *IEEE Transactions on Education*, Vol. 50, No. 3, August 2007.

27. D.J. Hwang and D.K. Blandford, "A Multidisciplinary team project for Electrical Engineering, Computer Engineering and Computer Science majors", *Proceedings of the 2000 ASEE Conference and Exposition*, St. Louis, MO, June 18 – 21, 2000.

28. E. Linde, D. Donlan, and M. Batchelor, "Mechatronics for Multidisciplinary Teaming", in *Proceedings of the 2003 ASEE Conference and Exposition*, Nashville, TN, June 22 – 25, 2003.

29. D.J. Ahlgren, I.M. Verner, I.M., D. Pack, and S. Richards, "Effective Practices in Robotics Education", in *Proceedings of the 2004 American Society of Engineering Education Annual Conference and Exposition,* Salt Lake City, UT, June 20 – 23, 2004.

30. D.J. Ahlgren, "Meeting Educational Objectives through Robotics Education", in *Proceedings of the 2002 5th Biannual World Automation Congress,* Volume 14, 2002, pp. 395 – 404.

31. J.A. Piepmeier, B.E. Bishop, and K.A. Knowles, "Modern robotics engineering instruction", *IEEE Robotics and Automation Magazine*, June 2003, pp. 33 – 37.

32. R.D. Beer, H.J. Chiel, and R.F. Drushel, "Using autonomous robotics to teach science and engineering", *Communications of the ACM*, June 1999, Vol. 42., No. 6, pp. 85 – 92.

33. H. Chaya, and G. Walker, "A 'real life' interdisciplinary capstone design course", in *Proceedings of the 2001 American Society for Engineering Education Annual Conference and Exposition,* Albuquerque, NM, June 24 – 27, 2001.

34. B. Ferguson, and D. Voltmer, "Engaging ECE Students in the practice of engineering", in *Proceedings of the 2006 American Society for Engineering Education Annual Conference and Exposition,* Chicago, IL, June 18 – 21, 2006.

35. J.O. Hamblen and T. S. Hall, "Engaging undergraduate students with robotic design projects", in *Proceedings of the Second IEEE International Workshop on Electronic, Design, Test and Applications (DELTA '04),* Issue 28 – 30, January 2004, 2000, pp. 440 – 442.

36. M.D. Heller, "Mobile Robots and Interdisciplinary design – MOBOTS", *Computers in Education Journal*, Volume 10, Number 1, pp. 73 – 76.

37. O. Hoffman, P. Dobosh, T. Djaferis, and W. Burleson, "Moving towards a more systems approach in a robotics based introductory engineering course at Mount Holyoke college", in *Proceedings of the 2005 American Society of Engineering Education Annual Conference and Exposition,* Portland, OR, June 12 – 15, 2005.

38. F. Rios-Gutierrez, and R. Alba-Flores, "Mobile robots capstone design course", in *Proceedings of the 2006 American Society for Engineering Education Annual Conference and Exposition*, Chicago, IL, June 18 – 12, 2006.

39. J.T. Tester, "Management of large team-design and robotics-oriented sophomore design class", in *Proceedings of the 38th ASEE/IEEE Frontiers in Education Conference*, Saratoga Springs, NY, October 22 – 25, 2008.

40. E. Wang, "Teaching freshmen design, creativity and programming with Legos and Labview", in *Proceedings of the 31st ASEE/IEEE Frontiers in Education Conference*, Reno, NV, October 10 – 13, 2001.

41. C. Kitts, "Surf, Turf, and Above the Earth", *IEEE Robotics and Automation Magazine*, September 2003, pp. 30 – 36.

42. B.C. Dickinson, O.C. Jenkins, M. Moseley, D. Bloom, and D. Hartmann, "Roomba Pac-man: Teaching autonomous robotics through embodied gaming", in *Proceedings of the 2007 American Association for Artificial Intelligence (AAAI) Symposium on Robot and Robot Venues: Resources for AI Education,* Palo Alto, CA, March 2007.

43. L. Greenwald, and J. Kopena, "Mobile Robot Labs", *IEEE Robotics and Automation Magazine,* June 2003

44. R. Manseur, "Development of an undergraduate robotics course", in *Proceedings of the 27th Annual ASEE/IEEE Frontiers in Education Conference*, Pittsburgh, PA, November 5 – 8, 1997.

45. B.A. Maxwell, and L.A. Meeden, "Integrating Robotics Research with Undergraduate Education", *IEEE Intelligent Systems,* November/December 2000, pp.22 - 27.

46. R. Meuth, P. Robinette, and D. Wunsch, "Introducing Robots", in *Proceedings of the 2009 American Society of Engineering Education Annual Conference and Exposition,* Austin, TX, June 14 – 17, 2009.

47. I.R. Nourbakhsh, "Robots and education in the classroom and in the museum: On the study of robots, and robots for study", in *Proceedings of the Workshop for Personal Robotics for Education, IEEE Internal Conference for Robotics and Automation (ICRA),* Piscataway, NJ. *2000.*

48. M. Rosenblatt, and H. Choset, "Designing and implementing hands-on robotics labs", *IEEE Intelligent Systems,* November/December 2000, Volume 14, Number 6, pp. 32 - 39.

49. A. Soto, P. Espinace, and R. Mitnik, "A mobile robotics course for undergraduate students in computer science", in *Proceedings of the IEEE Latin American Robotics Symposium (LARS),* 2006, pp. 187 – 192.

50. K.T. Sutherland, "Undergraduate robotics on a shoestring", *IEEE Intelligent Systems*, November/December 2000, pp. 28 – 31.

51. R. Siegwart., and I.R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, The MIT Press, Cambridge, MA, 2004, 335 pp.

52. M.J. Matarić, *The Robotics Primer,* The MIT Press, Cambridge, MA, 2007, 300 pp.

53. R.R. Murphy, *Introduction to AI Robotics*, The MIT Press, Cambridge, Massachusetts, 2001, 466 pp

54. Anon, Robotics Connection, Traxster II Robot base, http://www.roboticsconnection. com/, (accessed March 2010).

55. Anon, Microsoft Robotics Studio, http://msdn.microsoft.com/en-us/library/bb483024.aspx, (accessed March 2010)

Biographical Information

Dr. Berry is an assistant professor in the department of electrical and computer engineering at Rose-Hulman Institute of Technology. She is one of the principal investigators on the multidisciplinary educational robotics initiative and the Rose building undergraduate diversity program. Her research areas include the design and evaluation of human-robot interfaces and innovations in engineering education using active learning and mobile robotics.