Exposing Middle School Students to Robotics and Engineering through Lego and Matlab

Mr. Jeffrey Laut, Polytechnic Institute of New York University

Jeffrey Laut received his B.Sc. degree from the Polytechnic Institute of New York University in 2009 and his M.Sc. degree from Worcester Polytechnic Institute in 2011, both in Mechanical Engineering. He is currently a Ph.D. candidate at the Polytechnic Institute of New York University, where for the 2011-2012 academic year he was a teaching fellow in their GK-12 program. Laut conducts research in the Dynamical Systems Laboratory, where his interests include controls and robotics.

Dr. Vikram Kapila, Polytechnic Institute of New York University

Dr. Vikram Kapila is a professor of Mechanical Engineering at NYU-Poly, where he directs an NSF-funded Web-Enabled Mechatronics and Process Control Remote Laboratory, an NSF-funded Research Experience for Teachers Site in Mechatronics, and an NSF funded GK-12 Fellows project. He has held visiting positions with the Air Force Research Laboratories in Dayton, OH. His research interests are in K-12 STEM education, mechatronics, robotics, and linear/nonlinear control for diverse engineering applications. Under Research Experience for Teachers Site and GK-12 Fellows programs, funded by NSF, and the Central Brooklyn STEM Initiative (CBSI), funded by six philanthropic foundations, he has conducted significant K-12 education, training, mentoring, and outreach activities to integrate engineering concepts in science classrooms and labs of dozens of New York City public schools. He received NYU-Poly's 2002, 2008, and 2011 Jacobs Excellence in Education Award, 2002 Jacobs Innovation Grant, 2003 Distinguished Teacher Award, and 2012 Inaugural Distinguished Award for Excellence in the category Inspiration through Leadership. In 2004, he was selected for a three-year term as a Senior Faculty Fellow of NYU-Poly's Othmer Institute for Interdisciplinary Studies. His scholarly activities have included three edited books, six chapters in edited books, one book review, 51 journal articles, and 100 conference papers. He has mentored four doctoral students, eleven masters students, 25 undergraduate research students, and 11 undergraduate senior design project teams; over 300 K-12 teachers and 95 high school student researchers; and eighteen undergraduate GK-12 Fellows and 53 graduate GK-12 Fellows. Moreover, he directs K-12 education, training, mentoring, and outreach programs that currently enrich the STEM education of over 2,000 students annually.

Dr. Magued G. Iskander P.E., Polytechnic Institute of New York University

Dr. Magued Iskander is a professor of Civil and Urban Engineering at NYU-Poly. Dr. Iskander is a recipient of NSF CAREER award, Chi Epsilon (Civil Engineering Honor Society) Metropolitan District James M. Robbins Excellence in Teaching Award, Polytechnic's Distinguished Teacher Award, and NYU-Poly's Jacobs Excellence in Education Award (twice). Dr. Iskander's research interests include Geotechnical modeling with transparent soils, foundation engineering, and urban geotechnology. He makes extensive use of sensors and measurement systems in his research studies. Dr. Iskander has published ten books, 100 papers, and graduated six doctoral students, 27 masters students, twelve undergraduate research assistants, and supervised the research activities of three school teachers and nine high school students.

Exposing Middle School Students to Robotics and Engineering through LEGO and Matlab

1. Introduction

Use of robotics in the K-12 environment is becoming increasingly common to introduce students to educational and career opportunities in engineering and technology fields.[1] Moreover, several recent studies[2—4] have shown the effectiveness of robotics-based, hands-on science and math lessons *vis-à-vis* traditional classroom instructional practices. A wheeled mobile robot provides not only a compelling hook to engage students' interest, it also brings a new meaning to classroom math and science concepts by connecting them to real-world applications.

This paper considers a novel application of the LEGO Mindstorms platform and the RWTH-Mindstorms Toolbox for Matlab[5,6] to create a series of classroom activities that expose and engage middle school students in all aspects of controlling a differential-drive mobile robot. LEGO bricks are used to construct a mobile robot consisting of two non-steerable, powered wheels, and one castor wheel. The mobile robot uses a bluetooth connection to transmit wheel rotation data to a computer base-station running the Matlab program. An odometry model running on Matlab uses the wheel rotation data to keep track of the rotation and displacement of the robot on a Cartesian plane. This hardware-software platform is used to create and implement several lessons that illustrate real-world applications of concepts from middle school curriculum.

In the first lesson, the relationship between wheel rotations, wheel circumference, and distance traveled is derived. The effect of wheel diameter on forward velocity is assessed and the errors that may accumulate from the use of an incorrect diameter of the wheel are examined. The students use their knowledge of wheel rotations versus displacement to measure the length and width of the classroom. Next, the notion of feedback control is introduced through a controller that enables the robot to move a commanded linear distance. In the second lesson, the effect of commanding one wheel to move slower than the other wheel is observed. This leads to the derivation of the relationship for turning radius and the velocities of the individual wheels of the robot. Next, a complete odometry model is studied wherein feedback from the robot's encoders allows for the reconstruction of the robot's path in real-time. Using a joystick interfaced with the computer base-station, the students command the robot remotely over the bluetooth connection while the odometry data is transmitted from the robot to Matlab where the path taken by the robot is plotted in real-time. By driving the robot along the perimeter of the classroom and around obstacles, an occupancy-map of the classroom is created. As a final lesson, the so-called "parking problem" of the differential-drive robot is studied. The students are asked to think of a strategy for the robot to autonomously drive from a start position and orientation to a goal position and orientation. A controller based on the kinematic model of the robot is implemented

and demonstrated in class. The robot takes the goal position and orientation as input, and using odometry as feedback, smoothly steers to the desired pose. The performance of the feedback controller is compared to manual control where the students use the joystick to control the robot.

The above set of in-class lessons, demonstrations, and activities illustrate applications of mathematical concepts, primarily geometry and trigonometry, to solve practical problems in mobile robotics. Through these lessons, the students see practical value in subject areas that may otherwise appear to be abstract.

2. Motivation

Many students learn by doing. In every era, educators have used available technology to engage learners, e.g., abacus, maps, and electronic kits have been used to teach math, geography, and science, respectively. Although STEM concepts have been learned by past generations of students without robotics, we now live in an age in which most students are immersed in modern technology, even pre-schoolers have access to computers and video games, and they expect the ability to modify and control the experiment and receive immediate feedback. Thus, using primitive means for data collection commonly found in K-12 classrooms are not adequate for retaining the attention of students in contrast to the modern tools and techniques used by practicing scientists and engineers.[7] Moreover, the use of robotics in STEM learning offers youth a medium to apply their learning by doing activities that they find authentic, engaging, and meaningful. In fact, recent research has shown that the use of robotics-based activities helps to: (1) actively engage students in classroom instruction of K-12 math and science subjects and (2) increase students' understanding of the underlying science and math, thereby increasing effectiveness of lessons.[2—4]

In middle school math classes, students must learn abstract math concepts, graphical interpretation, problem-solving, and measurement methods. Research on *How People Learn*[8] suggests that effective teaching and learning strategies support students to see the connections between the science and math in their everyday lives and in the classroom. Many science and math principles are inherently incorporated into, reinforced through, and linked to students' daily experiences by performing simple tasks with a robot. The LEGO Mindstorms platform is a modular robotics platform that can easily be adapted to perform a variety of in-class lessons, demonstrations, and experiments to support learning in science and math disciplines. The wide variety of affordable sensors available, both from LEGO and third-party manufacturers, makes the platform even more attractive by offering the capability to measure temperature, acceleration, and light intensity, to name a few.

The lessons presented in this paper require students to actively explore concepts in math through their applications to the exciting field of robotics. For example, the students are asked to

employ their knowledge of geometry to determine the distance travelled by a robot. Such an exercise engages students in deeper learning through a real life experience in the use of relationship involving the diameter, circumference, and rotation of a wheel. Similarly, the students are asked to determine the distance between two locations visited by a mobile robot, thus providing them an experience with a contextual and practical application of the Pythagorean theorem.

The LEGO NXT's icon-based Mindstorms software environment, albeit user-friendly for K-12 students, is lacking in its ability to perform advanced calculations, execute intricate algorithms, and represent data in a flexible manner. RobotC, a programming environment that can be used to develop NXT programs using a C-like language, addresses some limitations of the Mindstorms environment. Unfortunately, RobotC helps to overcome Mindstorms environment's limitations only to a degree as allowed by the functions native to RobotC. That is, the RobotC environment does not integrate with other external C libraries and does not contain any data plotting capabilities. Moreover, the RobotC environment is only compatible with the Windows operating system and requires a purchased license.

Within the engineering community, Matlab is one of the most popular programming languages since it offers a multitude of functions and tools applicable to numerous engineering, science, computing, and math disciplines. Matlab's script-based programs are executed interpretively, making it very easy to develop and debug programs without the need to compile, and it is arguably an easier language to learn *vis-à-vis* C or Python.

The RWTH Mindstorms toolbox for Matlab,[5, 6] developed by RWTH Aachen University, provides a framework for real-time data acquisition from and control of individual or multiple NXT bricks over either a USB or bluetooth connection. Unlike RobotC, the RWTH Mindstorms toolbox is available for Windows, Linux, and Mac operating systems. While using the RWTH Mindstorms toolbox, the user also has access to Matlab's extensive plotting capabilities, which are extremely useful while using the LEGO NXT platform to engage students in conducting hands-on activities or performing classroom demonstrations. The use of Matlab along with LEGO NXT allows for a novel way to introduce middle school students to engineering through autonomous robots, while also exposing them to the programming, simulation, and graphing skills, that are valuable in themselves independent of other skills acquired through these hands-on activities.

3. Description

All of the lessons performed herein use the differential-drive mobile robot, shown in Figure 1. This type of robot uses relatively few parts, is easy to assemble, and has a high degree of maneuverability. A pair of motors drives each wheel and varying the relative velocity of each

motor allows for the turning motion of the robot. Each motor contains a built-in encoder that provides the angular position of the wheel, from which the position and orientation of the robot in a fixed frame can be derived. The wheel and robot velocity data can be obtained by using a finite-difference approximation of the encoder data. Additional sensors can be easily integrated on the robot platform through the use of any of the four available sensor ports on the robot's on-board computer, i.e., the LEGO NXT brick. The LEGO NXT brick has the aforementioned RWTH Mindstorms program installed on it and is configured to communicate bi-directionally, through a Bluetooth connection, with a laptop running Matlab (see Figure 2). The bluetooth connection permits wireless control of the robot within a range of approximately 30 feet. This long range allows the base computer, running the Matlab and associated toolboxes, to remain stationary while the robot is able to move around in a typical-size classroom.
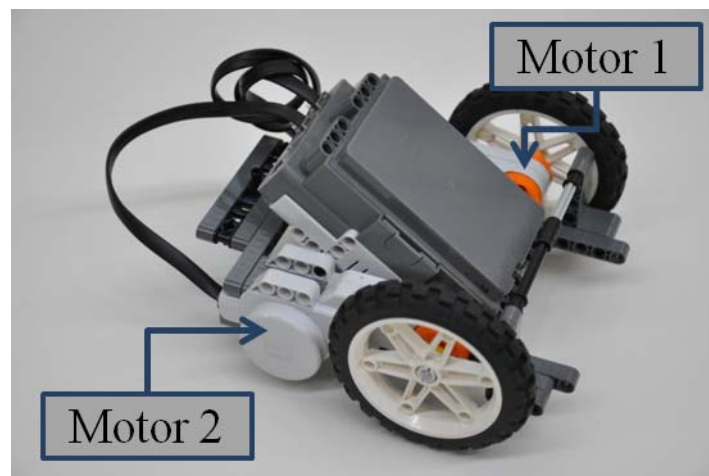

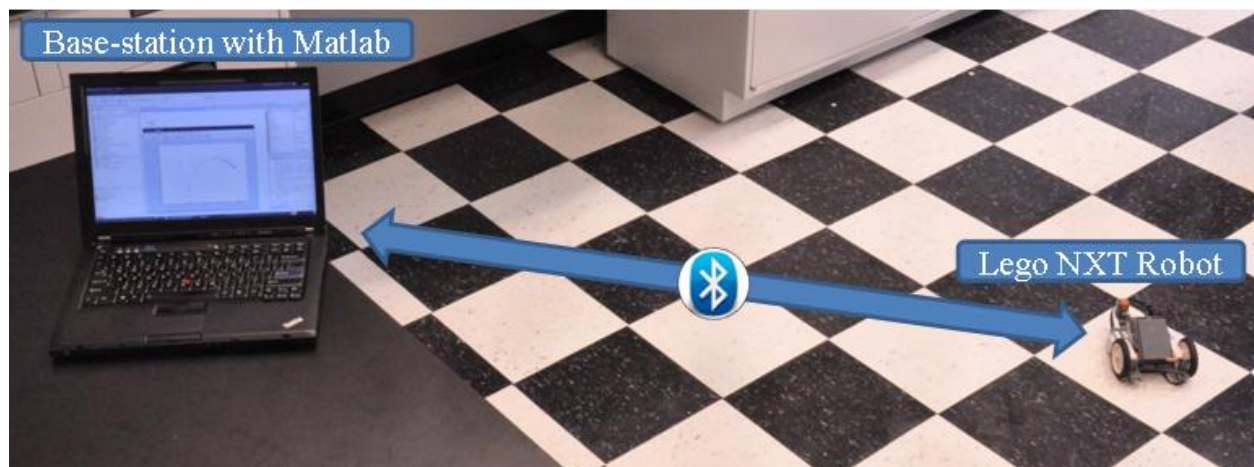**Figure 1:** A LEGO Mindstorms differential-drive robot.

Figure 2: A laptop base-station running Matlab and a LEGO robot communicate over a bi-directional bluetooth connection.

The aforementioned hardware, software, and communication platform is easily adaptable to command the robot to perform different tasks by simply executing suitable Matlab scripts. Such a framework helps illustrate to a young audience that although they see a *physical* mechanical device with embedded sensors, actuators, and computer, the underlying software is of equal, if not greater, importance in the proper functioning of the device. The following sections illustrate how to exploit the long-range communication capabilities and the ease of adaptability by using pre-written Matlab scripts to perform three entirely different lessons using the differential-drive mobile robot platform.

4. Lesson 1—Linear Distance from Wheel Rotations and Simple Feedback

Using the LEGO Mindstorms robot described above, Lesson 1: (1) reinforces the students' previously learned knowledge of circumference and ratios; (2) introduces to them the concepts of odometry and feedback control; and (3) allows them to practice their skills to measure distances. First, the students are assigned the task of measuring the diameter of a robot wheel using a standard ruler. Second, they measure the circumference of the wheel using a piece of string and a ruler. Third, the students observe that they can determine the circumference of any wheel if its diameter is known through the use of ratios, which leads to a discussion on the mathematical constant π. Fourth, a case study is considered to examine the effect of changing wheel diameter on the accuracy of the speedometer readings in an automobile. Fifth, the students follow up with a discussion on the distance a wheel travels if it makes one full rotation. Finally, through the above sequence of activities, the students discover that for each complete wheel rotation, the wheel travels the length of its circumference.

Having learned about how the wheel diameter, wheel circumference, wheel rotation, and distance traveled are related, the students are introduced to a function that takes number of wheel rotations as the input and provides distance traveled as the output. A Matlab script that implements this function is shared and discussed with the students. This basic script simply displays the distance traveled by the robot, which is commanded by a handheld gamepad and restricted to move along a straight line, while also displaying a visualization of wheel rotations and plotting the angular position of the wheel (Figure 3). Next, the students are asked if and how they can determine the wheel circumference provided that the following information is given: (1) the number of times the wheel rotates and (2) the distance traveled by the wheel. To concretize the students' understanding of these concepts through practice, the robot is programmed to perform a specific number of wheel rotations and stop after that. Next, using a tape-measure, the students determine the distance the robot has traveled, and in turn, the wheel circumference.

To introduce the topic of feedback control, the students are asked to brainstorm and come up with an "algorithm" for driving the robot a predetermined distance. Through guided discussion pertaining to the analogy of riding a bicycle, the students are led to discover a

proportional controller. Specifically, the students consider that while riding a bicycle if you are far from your goal you may want to ride quickly. Alternatively, as you get closer to your goal, you slow down until you reach your goal, at which time you stop. Next, a control algorithm is devised where the forward velocity of the robot is equal to a constant multiplied by the distance to the goal. This algorithm is implemented in a Matlab script, which is shared and discussed with the students. The script is then executed and the students observe, examine, and critique its performance.
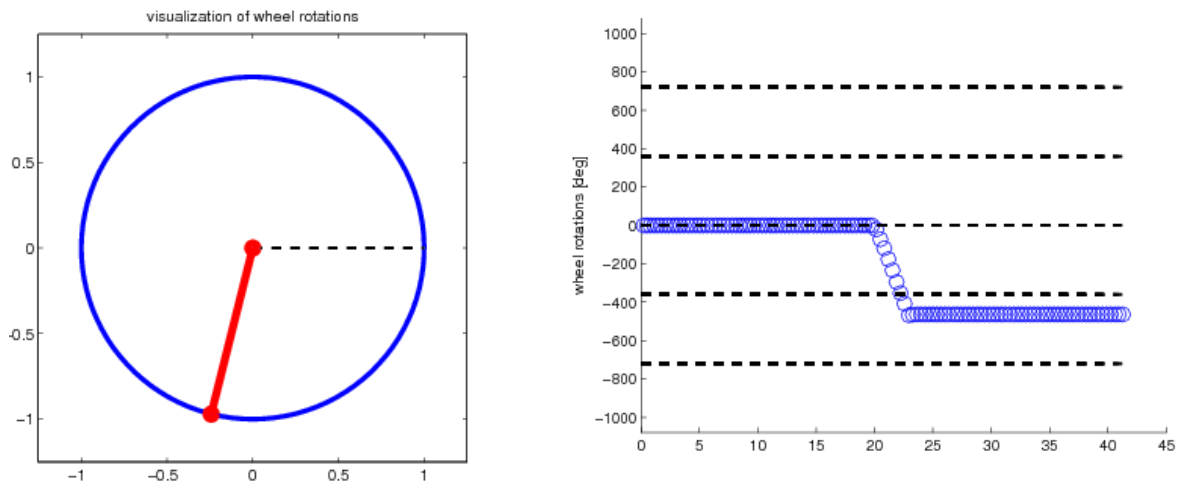
Figure 3: Real-time visualization of odometry data. Here, the wheel has rotated approximately one and a quarter turn. (a) Dashed line indicates the wheel start position and red line indicates current position and (b) angular position of the wheel in degrees.

5. Lesson 2—Classroom Mapping

In this activity, we build upon the basic odometry model of the previous lesson and investigate the effect of varying one wheel's velocity relative to that of the other wheel on the turning radius of the robot. A complete odometry model is then implemented on the robot, allowing for creating a map of the classroom as the robot moves.

Since the encoders on the motors provide only angular position data, a numerical derivative is used to approximate the angular velocity of each wheel. Although in practice a second-order backward difference approximation is commonly used to obtain an accurate estimate of the velocity, in this lesson, we introduce to students the concept of a derivative using a first-order approximation, wherein a change in position divided by a change in time provides an estimate for the velocity. To demonstrate the numerical derivative in class, the mobile robot traverses a known distance, which is measured with a yardstick, at a constant speed. Using stopwatches, the students time how long it takes the robot to traverse the distance, after which they divide the measured distance by the measured time to estimate the average velocity.

To investigate the effect of wheel velocity on turning radius of the robot, we consider the following expression

$$R = \frac{l}{2}\left(\frac{V_r + V_l}{V_r - V_l}\right),$$

where R is the instantaneous center of curvature of robot's path, l is the track width of the robot, i.e., the separation distance between the right and the left wheels, and V_r and V_l are the right and left wheel velocities.[9] The students are not asked to derive this equation, instead it is simply presented and explained to them. The students then discuss some special cases. For instance, when both wheel velocities are equal, R is infinite, which means that the robot is traveling in a straight line. When the wheel velocities are equal in magnitude, but opposite in direction, R is equal to zero, which means the robot is turning in place. If V_r or V_l is zero, then R is equal to half of the track width of the robot. Each of these cases is then programmed to be executed on the robot, allowing the students to observe and confirm the validity of the above equation in these scenarios.

To make full use of the encoder data, a complete odometry model can be implemented to provide the absolute position and orientation of the robot in a fixed reference frame. The complete odometry model is characterized by

$$x(t) = \int_0^t V(t)\cos(\theta(t))dt, \quad y(t) = \int_0^t V(t)\sin(\theta(t))dt, \quad \theta(t) = \int_0^t \omega(t)dt$$

where $x(t)$, $y(t)$, and $\theta(t)$ represent the x-position, y-position, and angular orientation, respectively, of the robot in a fixed reference frame, $V(t)$ denotes the forward velocity of the robot, and $\omega(t)$ represents the angular velocity of the robot. Using this odometry model, the mobile robot can be driven remotely via the Bluetooth connection while its absolute position is plotted using Matlab plotting routines in real-time. Such a framework allows the robot to be used as a tool for mapping an environment. Figure 4 illustrates the schematic of a control system used for classroom mapping. As a final activity for this lesson, the students are asked to use the Pythagorean theorem to find the distance between two locations given by the (x_1, y_1) and (x_2, y_2) coordinates in the classroom mapped by the robot.
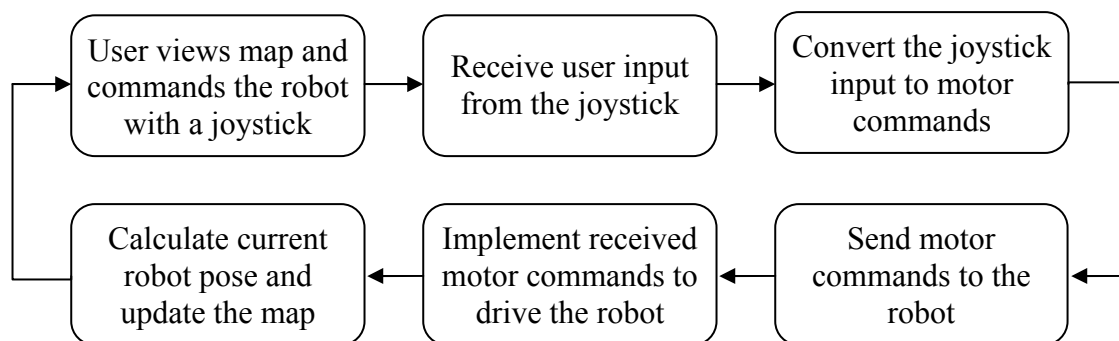

Figure 4: Schematic of control system used for classroom mapping.

6. Lesson 3—Autonomous Motion

In this activity, we further build upon the odometry model of the previous lesson and implement a nonlinear controller to implement closed-loop steering. While in the previous lesson, a user controlled the robot remotely, in this lesson, the robot uses its current position in the fixed frame as feedback for a position controller that autonomously parks the robot. The forward and angular velocities are computed by

$$V = (\gamma \cos\alpha)e, \quad \omega = k\alpha + \gamma \frac{\cos\alpha \sin\alpha}{\alpha}(\alpha + h\varphi)$$

where γ, k, and h are positive control gains, e is the magnitude of the error vector e representing the goal position with respect to the robot's current position, φ is the angle of e with respect to the goal orientation, and α is the difference between φ and θ, the robot's orientation with respect to the goal orientation (Figure 5).[10]
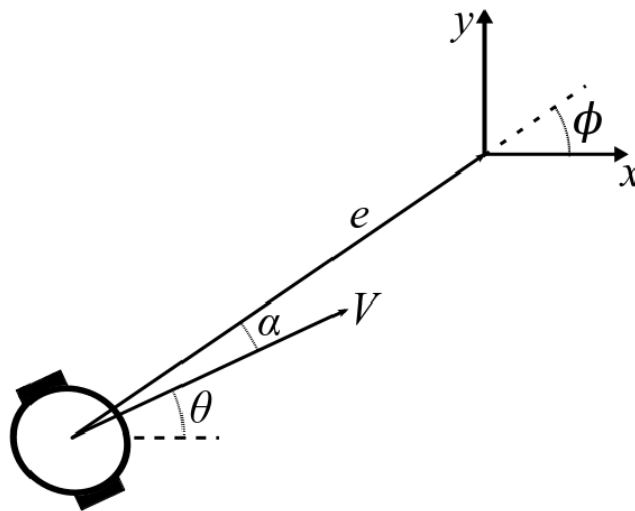

Figure 5: Position and orientation of a LEGO robot with respect to goal position and orientation.

Through this lesson, the students are introduced to trigonometric functions and taught how the controller operates by iterating over the same equations. First, we introduce the need for trigonometric functions by asking the students to measure the x- and y-components of a right triangle by using a ruler. Second, we ask the students to brainstorm a quicker way to do this since our robot needs to operate quickly and it does not have the ability to explicitly measure in the x- or y-directions. Third, we introduce two "black boxes," both of which take a length and an angle as inputs, and one black box outputs the x-component while the other black box outputs the y-component. We later name these black boxes trigonometric functions "cosine" and "sine," and use them to develop our controller. The programming script is shared and discussed with the students to expose them to a real programming scenario where such functions are used to perform a specific task, for instance, autonomously parking a robot. Finally, we discuss the iterative nature of the resulting program, wherein the computer performs the same calculations

rapidly until some criterion is reached to exit a loop. The purpose of this exercise is to make the students aware of how a computer program iteratively executes the equations described. The effect of adding a pause within the main loop of the program also helps to illustrate the need for efficient code and fast processing power to have the system perform satisfactorily. Figure 6 below illustrates the control systems corresponding to the autonomous motion case. Finally, Figure 7 illustrates the performance of the automatic control system for the autonomous parking problem.
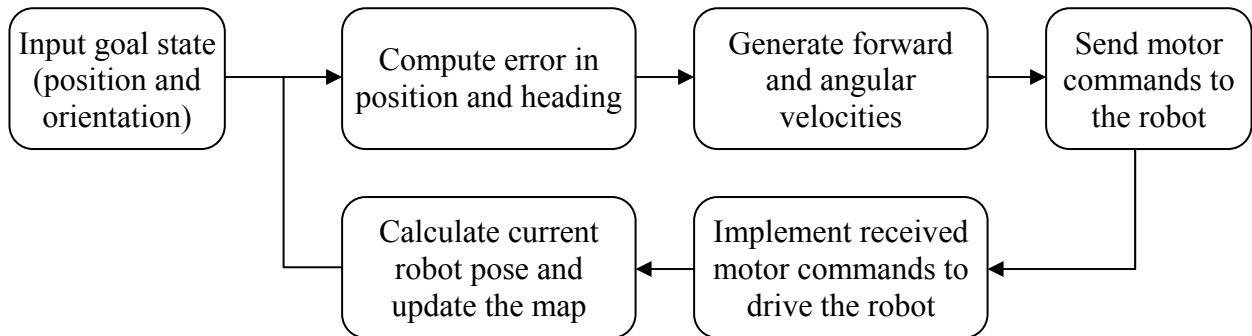
Figure 6: Control system diagram for autonomous motion of the robot.
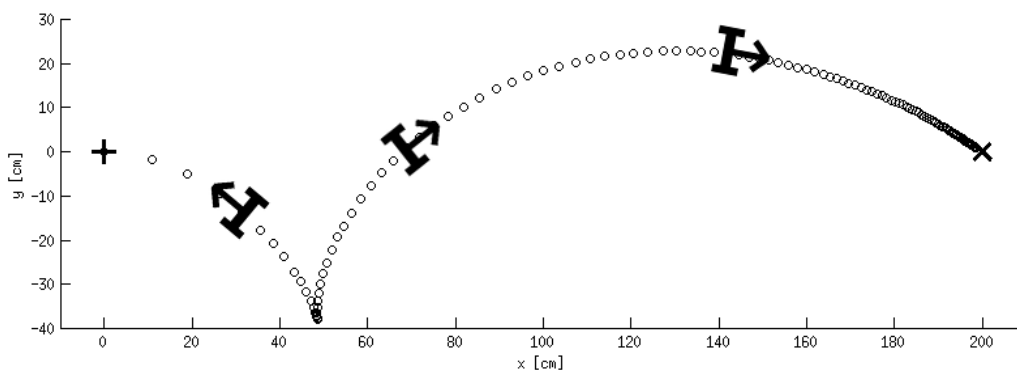

Figure 7: Odometry data results for autonomous parking. Plus symbol indicates the start state (0, 0, 180°), cross symbol indicates the goal state (200, 0, 0°), and circles represent instances of odometry data. The arrows on the robot symbol point towards the front of the robot. From the start position, the robot executes a backward turning motion. At the coordinate position (48.6, −38), the robot completes the backward turning motion and begins the forward turning motion.

7. Classroom Evaluation Method

For each lesson, introductory concepts are presented to the students prior to engaging them with LEGO-based activities. The aforementioned lessons and activities were conducted
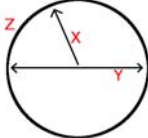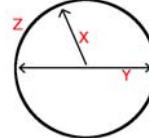
over two consecutive 45-minute class periods. To gauge the effectiveness of LEGO- and Matlab-based activities in enhancing the lessons, students' understanding of lesson concepts is examined through pre- and post-lesson assessment surveys. The first 15 minutes of the first class period were used for the pre-assessment and the last 15 minutes of the second class period were used for the post-assessment. To facilitate the evaluation of specific topics, questions that are fundamentally identical yet worded differently are included on both the pre- and post-lesson assessments. For example, to assess students' understanding of the Pythagorean theorem, the pre-lesson assessment asks the students to find the hypotenuse c of a right triangle with sides a and b. Alternatively, on the post-lesson assessment, the problem is reformulated and asks the distance between the start and goal locations of a robot that travels from the start point 4 feet in the x-direction and then 3 feet in the y-direction to arrive at its goal point. Table I provides a listing of the four content questions from the pre- and post-lesson assessments. Next, the results of the pre-and post-lesson assessments are examined as follows. First, for each analogous pair of questions, the responses are graded as either "correct" or "incorrect." Second, for each content question, pre- and post-lesson class average is determined. Third, to analyze the statistical significance of the assessment results, a dependent t-test for paired samples is conducted. Finally, for the autonomous motion lesson, the students used a joystick to attempt to park the robot and compared this experience *vis-à-vis* the observed performance of the robot under the control of the autonomous motion algorithm as depicted in Figure 7.

8. Data Analysis

The aforementioned lessons and assessments have been conducted in two 6th grade classrooms (one with 13 students and the other with 21 students) in two middle schools in Brooklyn, NY. As seen through Figure 8, average class score on each content question (Q1—Q4) increased from the pre-assessment to post-assessment. Moreover, taken as a whole, the pre- and post-lesson assessments show a general increase in the class performance following the robotics activities.

Finally, a t-test for paired samples is conducted to evaluate the effectiveness of the robotics-based lesson on improving the students' grasp and understanding of the concepts introduced through the robotics activities, see Table II. This evaluation was performed by using the students' responses to pre- and post-assessment individual content questions and their average scores. As seen through Table II, for Q2—Q4 and students' average scores, the results of t-tests reject the null hypothesis that there was no change in the students' performance using a 2%, or better, significance level. That is, the students' performance from pre- to post-test increased significantly and we can state with a confidence level of 98%, or higher, that the robotics-based activities played a significant role in this gain.

Table I: Content questions on pre- and post-lesson assessment surveys.

| Pre-assessment content questions	Post-assessment content questions
Q1. Which letter in the figure presented is the diameter (circle one)? X Y Z 	Q1. Which letter in the figure presented is the circumference (circle one)? X Y Z
Q2. If a circle has a radius of r centimeters, how do you find its circumference using π?	Q2. If a circle has a diameter of D centimeters, how do you find its circumference using π?
Q3. What is the length c of the hypotenuse of a right triangle if the other two sides are of lengths a and b (circle one)? i) $a+b$ ii) $\sqrt{a+b}$ iii) a^2+b^2 iv) $\sqrt{a^2+b^2}$ **Alternative version** A person walks 3 meters in the x-direction, then 4 meters in the y-direction. How do you find how far the person walked (circle one)? i) $3+4$ ii) $\sqrt{3+4}$ iii) $\sqrt{3^2+4^2}$ iv) 3^2+4^2	Q3. A robot travels 4 feet in the x-direction, then 3 feet in the y-direction. How do you find its distance from the starting point (circle one)? i) $4+3$ ii) $\sqrt{4+3}$ iii) 4^2+3^2 iv) $\sqrt{4^2+3^2}$
Q4. Given that a circle has a diameter D, what is its circumference C (circle one)? i) $\dfrac{D}{\pi}$ ii) $2\pi D$ iii) πD iv) $\dfrac{D}{2\pi}$	Given that a circle has a circumference C, what is its radius r (circle one)? i) πC ii) $\dfrac{C}{2\pi}$ iii) $\dfrac{C}{\pi}$ iv) $2\pi C$

(a)

(b)
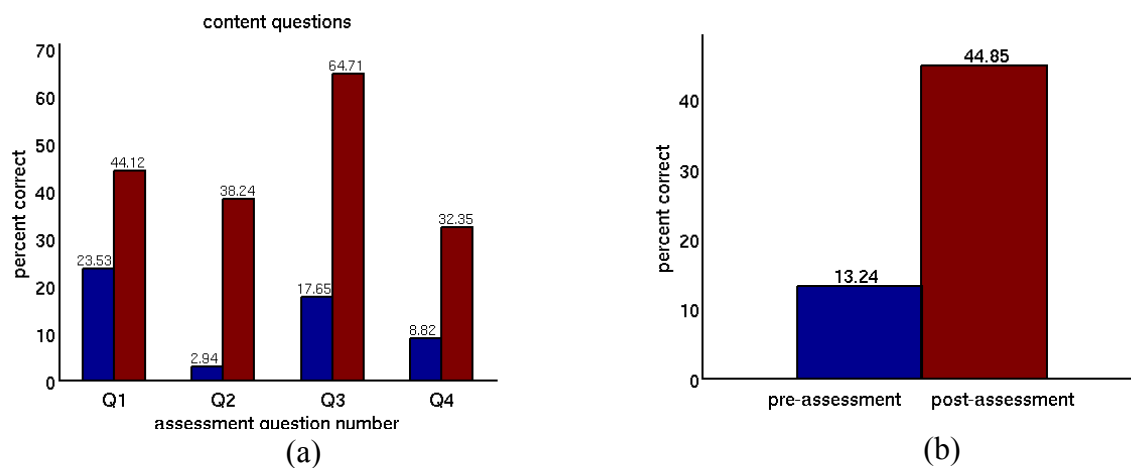
Figure 8: Performance of class before and after activity: (a) class average for individual questions and (b) class average for the entire assessment.

Table II: Results of a dependent *t*-test for paired samples. Values calculated using students' responses to each individual content question as well as on students' average scores, with *n*=34.

Item	Mean Difference	Standard Dev.	*t* calculated	*p* value
Q1	20.6	59.2	2.02	Not significant
Q2	35.3	48.5	4.24	< 0.0002
Q3	47.0	50.7	5.42	< 0.0001
Q4	23.5	55.4	2.48	< 0.02
Aver.	31.6	24.9	7.42	< 0.0001

For the autonomous motion lesson, the following qualitative observations can be made. After conducting this lesson, many students commented on the level of difficulty in using the joystick to perform the given parking task versus the fluent motion of the robot under the autonomous control algorithm. This shows that the students gained an appreciation for the role of computer algorithms in a real-world application situation.

9. Conclusions

We used a LEGO NXT-based robot communicating with a base-station running Matlab to demonstrate the autonomous capabilities of mobile robots to middle school students. In building the foundations for autonomous motion, various mathematical concepts typically taught in middle school were addressed through demonstrations and student-explorations. Pre- and post-assessments showed a statistically significant increase in students' knowledge through the robotics-based lessons. This suggests that even as traditional classroom instruction introduces fundamental concepts to students, use of robotics-based activities can tie these concepts to real-world engineering examples, which may offer authentic contexts in which students can relate and apply the newly learned material. Such an approach can develop students' abilities in solving real-world problems related to their classroom learning. Finally, middle school curricula typically lack exposure to engineering and technology. Introducing young students to engineering and technology may offer a means to engage students who may otherwise be uninterested in science or math, but find interest in applying the science and math concepts to engineering and technological problems.

To increase the number of middle school students who receive exposure to robotics and engineering, it is necessary to allow their teachers to develop and hone their skills to perform robotics-based activities. In this spirit, a series of demonstrations introduced some of the aforementioned activities to over 10 New York City middle school teachers during a *Teaching STEM with Robotics Workshop* held in summer 2012 at the Polytechnic Institute of NYU. In a follow-up workshop, planned for summer 2013, participating teachers will be engaged to individually perform all of the aforementioned activities to enhance their skills and confidence.

Finally, to reach out to a broader audience of teachers, we will prepare and submit an activity lesson plan for publication and dissemination through the Teachengineering.org website of the National Science Digital Library.

Acknowledgements

References

1. Rogers, C. and Portsmore, M. 2004. "Bringing Engineering to Elementary School." *Journal of STEM Education*. 5(3): 17-28.
2. Norton, S. 2004. "Using Lego to Integrate Mathematics and Science in an Outcomes Based Syllabus." *Proc. AARE Annual Conference*. Melbourne, Australia.
3. Ortiz, A.M. 2011. "Fifth Grade Students' Understanding of Ratio and Proportion in an Engineering Robotics Program." *Proc. Amer. Soc. Eng. Ed.* Session M444B. British Columbia, Canada.
4. Williams, K., et al. 2012. "Enriching K-12 Science and Mathematics Education Using LEGOs." *Advances in Engineering Education*. 3(2).
5. RWTH - Mindstorms NXT Toolbox for MATLAB. Online: http://www.mindstorms.rwth-aachen.de/.
6. Behrens, A., Atorf, L., and Aach, T. 2010. "Teaching Practical Engineering for Freshman Students Using the RWTH - Mindstorms NXT Toolbox for MATLAB." In *Matlab-Modelling, Programming and Simulations*. Pereira Leite, E. (ed.). 41—65. InTech. ISBN 978-953-307-125-1.
7. Yakubov, N., et al. 2005. "Integration of Real-Time Sensor Based Experiments in High School Science Labs: A GK-12 Project." *Proc. Amer. Soc. Eng. Ed. Session* 1510. Portland, OR.
8. NRC. 2000. *How People Learn: Brain, Mind, Experience, and School*. Bransford, J.D., Brown, A.L., and Cocking, R.R. (eds.). National Research Council (NRC). Washington, DC: National Academy Press. Online: http://www.nap.edu/catalog.php?record_id=9853.
9. Dudek, G. and Jenkin, M. 2010. *Computational Principles of Mobile Robotics*. Cambridge University Press.
10. Aicardi, M., et al. 1995. "Closed Loop Steering of Unicycle like Vehicles via Lyapunov Techniques." *IEEE Robotics and Automation Magazine*. 2(1): 27—35.