IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received February 16, 2017, accepted March 17, 2017, date of publication March 30, 2017, date of current version May 24, 2018.

Digital Object Identifier 10.1109/ACCESS.2017.2689686

A Versatile Approach for Teaching Autonomous Robot Control to Multi-Disciplinary Undergraduate and Graduate Students

AIDAN F. BROWNE, (Senior Member, IEEE), AND JAMES M. CONRAD, (Senior Member, IEEE)
University of North Carolina at Charlotte, Charlotte, NC 28223-0001, USA

Corresponding author: A. F. Browne (aidanbrowne@uncc.edu)

ABSTRACT Learning autonomous vehicle programming can be very challenging for students, especially when combined with robotic vehicle design and construction. This paper presents a methodology successfully used over the past six years to teach autonomy using a versatile platform built upon a commercially available product. A number of courses have been taught using the methodology at both the undergraduate and graduate levels. Students' ability to successfully learn and produce a solution to an autonomous robot control challenge has shown to be very effective.

INDEX TERMS Educational robots, motion control, sensor fusion, autonomous agents.

I. INTRODUCTION

A set of upper-division/graduate-level courses have concentrated on electrical, computer, and mechanical topics related to autonomous robotic vehicles rather than traditional manufacturing robotics. Topics include the theoretical underpinnings of robotics technologies validated with hands-on robotics laboratory exercises.

A. OTHER COURSES WITH AUTONOMOUS VEHICLES

The use of autonomous vehicles has become prevalent as a tool for teaching mechatronics, programming and complex control. Carnegie Mellon's Robot Autonomy course set the framework for using creative tasks to spark the interest and learning of students as early as 2002 [1], [2]. The LEGO MindStorms NXT is a low cost platform in very wide use that rapidly expanded the ability to quickly produce autonomous vehicles based on canned libraries and a powerful graphical programming language [3]; we also use it in our courses. Competitions also have proven to be a valuable tool in motivating students to advance their ability to program a robotic vehicle to act autonomously [4], [5]. There are many other examples of successful approaches to teaching autonomy.

B. SIMILAR PLATFORMS FOR TEACHING AUTONOMY

The platforms used to teach autonomous control are as varied as the curricula for teaching, and vary across a broad range of capabilities and price points [6]. The choice of platforms also cross the spectrum of robustness from toys, to hobbyist level to industrially robust. The controllers selected for the robots also vary across a similar spectrum from custom boards, to hobbyist level electronics such as Arduino [7], to industrial level platforms such as the National Instrument sbRIO or myRIO. While some academic programs choose to favor commercially available platforms [8], others prefer to develop their own to specifically meet their specific needs [9], [10].

II. OVERVIEW

The content of this laboratory series has been designed for a multi-disciplinary group of students made up for undergraduate and graduate students from Electrical and Computer Engineering, Mechanical Engineering, Engineering Technology and Electromechanical Systems. By nature of the varied backgrounds, many topics need to be introduced or reviewed that may be taken for granted in a classical Electrical Engineering or Computer Engineering curriculum. Each student is required to have already completed a programming and an electronics course. The use of the series has been expanded, and is now used in two departments. Depending on the specific course the series is taught within, the students are second-semester juniors, seniors, or graduate students. Because of the varied multi-disciplinary backgrounds, the same expectations were applied to both undergraduate and graduate students.

Hands-on labs are a major component of the series. Many others have made evident the positive effect that interacting with a complex plant to complete tasks has on a student's ability to motivate themselves to learn. The students worked in groups throughout the series to leverage the benefits of group learning.

The objectives of the experiment series were to:

1) Program a modern industrially significant microcontroller using LabVIEW and operate its peripheral devices.
2) Become familiar with industry significant sensors and learn how to interface with them.
3) Understand basic concepts of locomotion using a wheeled robotic platform and design control of motion sub-systems (DC motors, encoders, servos).
4) Design a motion trajectory planning algorithm and implement it in a robot.
5) Identify general concepts of Systems Engineering.

In our past experience, many attempts to have students build and program an autonomous robot end up unsuccessful because students tend to dedicate the majority of their time to the robot design and assembly; they frequently underestimate the level of complexity of the task of actually getting a physical plant to respond as expected. One of the approaches in the development of this series was to remove the robot design and mechanical assembly from the process, allowing students to focus on integration of environmental sensors and characterization of vehicle response.

It has been our experience that students do not really appreciate the need to fuse sensor inputs in order to allow a robotic vehicle to accomplish a task. In their purist theoretical manner of approaching the problem, they tend to assume all ideal conditions. To let them self-discover the need for incorporating closed loop feedback, the first experiment performed is to drive the robot repeatedly along the perimeter of a 2 meter square taped on the floor. The task is achievable with a dead-reckoning solution, but the students quickly get frustrated with the non-repeatability of their solutions. Factors such as inconsistent frictional forces between the left and right drive trains and wheel slippage lead them initially to complain that the robot doesn't work correctly; they are then taught that these are real-world vehicular challenges that need to be overcome. Students are subsequently taught about quadrature encoders and given the option to incorporate them in their designs. The improved ability for the robot to complete the squares perimeter without drift has been observed to nudge the students in the direction of being sold on the value of using closed loop sensor feedback to compensate for some of the previously unexpected phenomena.

The remainder of the series introduces the students to each of the sensors, and has them write a program to manipulate the robot to maneuver in response to the sensors. These will be discussed after first giving an overview of the robotic system. A canned-solution approach is not used; instead, the students are given data sheets for the industrial sensors and motors that are mounted on the platform, and expected to learn how to interface with them, both electrically and programmatically. This hierarchical approach helps the students understand the value of breaking a complex task into sub-tasks, and allows the introduction of project management skills in real-time while working on an actual complex project.

III. ECOSYSTEM

The platform we originally implemented for teaching autonomous robot control has been constantly evolving. After early experiences using Gears articulating platforms and Digilent robotics kits, in 2011 we switched to using the NI LabVIEW Robotics Starter Kit for Prototyping (a.k.a. DaNI). The construction of the platform made it very easy to add additional features and modify the functionality of the platform. Originally the platform was based on the National Instruments sbRIO family of control boards, and we recently up-fitted a group of the platforms to run off of the National Instruments myRIO, greatly expanding the versatility and programmability. The documentation guiding the students through the process has also evolved over time in synchrony with the hardware changes.

A. HARDWARE PLATFORM

The first two times the course was taught, students created autonomous robotic vehicles starting from a mostly solved mechanical platform. This platform was a four-drive wheel, articulated-axle vehicle capable of travel on rough terrain. Students added a battery and power distribution subsystem, motor driver circuitry, sensors, and a microprocessor-based controller. While most student groups were able to implement a working vehicle, the vehicle did nothing more than travel forward, backward and turn in order to maintain a six-foot distance from a walking human.

Students were never able to explore more interesting topics like localization, mapping, and navigation because they were spending so much time on subsystem activities like circuit assembly, circuit debugging, and device driver programming. Also, every platform constructed was unique, and often rather fragile. While the base mechanical structure could be reused, the electronics had to be rebuilt each semester.

The need for a more robust platform led to the specification and purchase of eight DaNI 1.0 robots. Originally we chose the platform because it was industrially relevant, based on the National Instruments sbRIO-9631. As detailed in the next section, we augmented the platform with additional sensors; we designated the upgraded units DaNI 1.1 to distinguish them from the original configuration; it is shown in Fig. 1.

One difficulty with the platform was its resistance to smooth turning, due to its high center of gravity and its narrow four-rubber-wheel base. NI and Pittsco upgraded the design to a DaNI 2.0 in 2011, by lowering the chassis, replacing the two rear wheels with a single omni-wheel and upgrading to the sbRIO-9632. We purchased eight of these next generation robots in 2012. After having made the same sensor additions to the 2.0 as we did on the 1.1; for clarity, we refer to this configuration as the DaNI 2.1; it is also depicted in Fig. 2.

In 2014 we gave graduate students in a Mechatronics course a project of physically upgrading the older DaNI 1.1

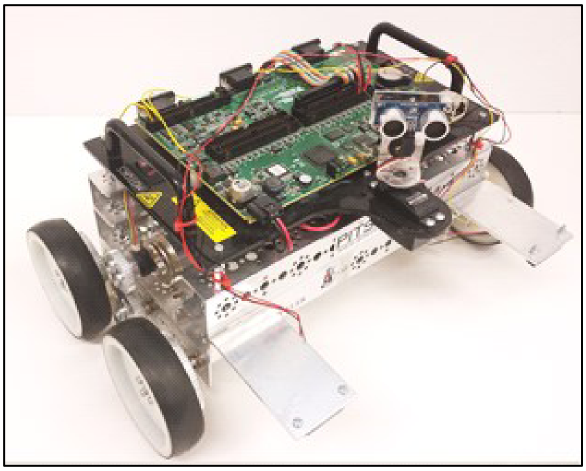

FIGURE 1. DaNI 1.1.

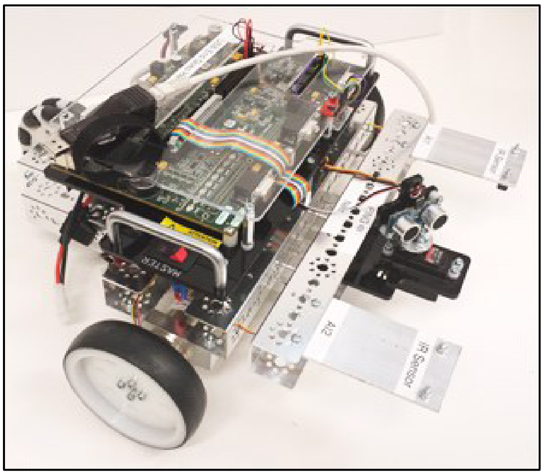

FIGURE 2. DaNI 2.1.

platforms by reconfiguring the existing aluminum structural members and adding the same rear omni-wheel in place of the two rear rubber wheels [11]. We designate these physically equivalent to DaNI 2.1 robots, with the older sbRIO, Dani 1.5 robots.

With the expanded fleet of DaNIs, we have now have the ability to make the robots available to multiple courses simultaneously, as well as for student projects. The flexibility of the platform has allowed up to use it over an extended period of time, and give students the experience of integration and functional design of a robot, without the past challenges of mechanical design and assembly of a complex vehicle. In a later section we discuss upcoming modifications to the platform, including vision navigation and LIDAR.

B. SENSORS

The sensor suite on the DaNI 1.5 and 2.1 platforms give the students exposure to industrially significant sensor technologies, and give them the minimum set of tools they need to accomplish the mission of successful autonomous robot navigation.

Natively both platforms include quadrature encoders on each of the driven wheels. Implementation of these sensors within the closed-loop control of the robot movement allows for much more accurate results by allowing the students to compensate for frictional differences in the drive trains, as well as wheel slippage.

The other natively included sensor is a servo-rotatable ultrasonic 2 cm. to 3 m. distance sensor, the Parallax PING sensor. FPGA code scans the sensor in a 180 degree arc while continuously measuring distance to the nearest object. This feedback allows the students to write logic to make decisions based on the dynamic ''map'' of the area immediately in front of the robot. Some students have decided to customize the FPGA code to only give measurements at discrete points (e.g. −90, 0 and 90 degrees), for the purposes of decreasing the mapping update time.

One of the sensors added in the DaNI 1.1 / 2.1 upgrades was a two axis ADXL320 accelerometer. This low cost single monolithic IC based sensor gives a signal conditioned analog voltage output proportional to sensed acceleration. Because it can measure both dynamic and static acceleration, it can be used to sense the gravity vector, and therefore be used as a tilt sensor. This is useful for students when navigating the obstacle course, as it can let them ''teach'' the robot when it is on a ramp or bump.

The last sensor that we added in the DaNI 1.1 / 2.1 upgrades was a pair of down-looking infrared sensors, Sharp part number GP2Y0A21YK. Brackets were added to the front left and right corners of the robot platform to ensure the sensors were forward of the wheels. The sensors can be implemented by the students as cliff sensors, allowing the robot to determine that it is about to drive off of an edge, and therefore save itself.

The use of each of these sensors is introduced to the students in a discrete laboratory exercise. They are then encouraged to use them as the building blocks for developing logic to allow the robot to autonomously move though the obstacle course maze. Students are encouraged to use sensor fusion to increase the reliability of their developed logic's decision making.

C. SOFTWARE

The choice of the DaNI platform allows the use of LabVIEW as the programming language, which has been shown repeatedly to be an excellent rapid-prototyping software platform [12]. Students are able to program at the higher functional block level, which keeps them from getting lost in the details of programming the low level code.

With the upgrade of the controller to an NI myRIO, the ability to program in C was also added. This gives much more flexibility with the platform, allowing it to be used in other courses. Whichever programming language is used, students are encouraged to design their logic before beginning the coding process. A sample of one student team's flowchart is shown in Fig. 3.

Both the sbRIO and myRIO controller platforms are part of NI's RIO family, which stands for Reconfigurable

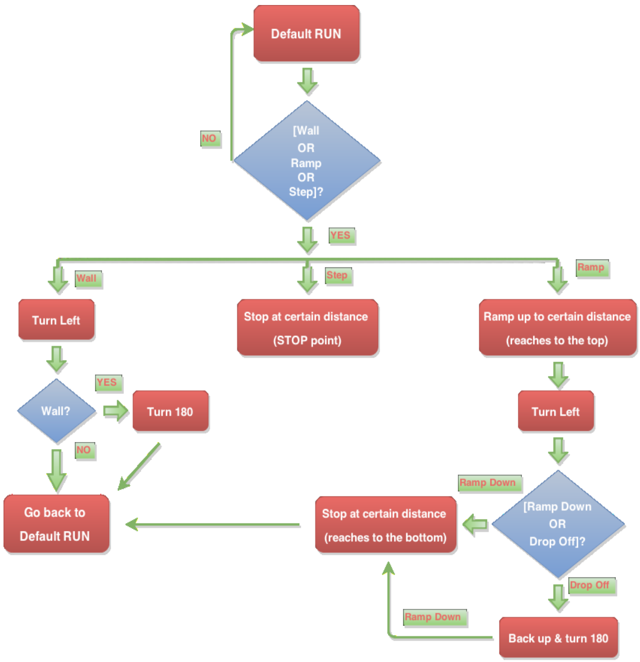

FIGURE 3. **Student developed logic for navigating course.**

Input/Output. All of the device pins connect to an FPGA which allows the front end pre-processing of the signal to be customized. This has opened up the opportunity of allowing students to program an FPGA, without having the deep knowledge normally required to program in VHDL. LabVIEW FPGA allows students to program in the familiar LabVIEW interface, and have the FPGA code converted to VHDL and compiled in the background. This greatly shortens the amount of time required.

D. EDUCATIONAL APPROACH

A 2010-2011 Senior Design team was charged with creating up to seven laboratory assignments that the Robotics students could use to implement more complex robot tasks like search and rescue and travelling in a convoy. These Senior Design students worked with the faculty member to specify the lab assignment content, write the lab exercises, and solve the lab exercises, including recording videos of the operating vehicles [13]. National Instruments was interested in these lab exercises and funded a student to port them to the DaNI 2.0 Platform in the summer of 2011. The results of this work was presented at the August 2011 National Instruments NI Week Conference.

The following lab exercises using the National Instruments DaNI robotic vehicle have been used over the last six years with over 190 Electrical, Computer, and Mechanical Engineering students. The laboratory exercises are each supported with background information from *Introduction to Autonomous Mobile Robots* [14], which is used as a required text for one of the three courses.

1. Introduction: The purpose of this experiment is to familiarize the student with LabVIEW and to introduce the robot that they will be working with in the class. Students learn some of the nuances of LabVIEW and use tutorial process to make the DaNI vehicle "roam" around and not hit walls or objects.

2. Motor Control: The purpose of this experiment is to introduce the concepts of motor control and to demonstrate this by programming DaNI to complete a 2 by 2 meter square. Students set up a loop that will step through the process of having the robot move in a square path and then stop once it completes the path. Some students' solutions use only timing to complete the squares, but some groups also use the on-board wheel encoders to make more precise movements.

3. Edge Detection: The purpose of this experiment is to introduce the student to hierarchy within LabVIEW by programming edge detection and still allowing concurrent obstacle avoidance behavior. To accomplish the goal of edge detection two analog-based Sharp infrared (IR) sensors are used. The sensors are able to easily distinguish the difference between a floor 5cm. away and the absence of the floor (like just over the edge of a stair).

4. A(star) Path Planning: The purpose of this experiment is to introduce A* path planning and implement the algorithm with the robot to navigate a room. The goal of the algorithm is to analyze the surrounding area of the object at each point along the way to the desired destination in order to obtain the shortest path while avoiding obstacles. This is a good introduction on localization, mapping, and navigation.

5. Obstacle Course: This experiment is used to bring together all of the concepts that the students have learned throughout the semester. In this experiment, there is an obstacle course the vehicle must traverse and the students may use any method they wish. The methods that may be used in this experiment include obstacle avoidance, A* Path Planning, motor control, or sensing and control. The students may also use the accelerometer or any other available sensors.

In order to be deemed successful, students must have their robots navigate a maze constructed within a 10 ft. by 10 ft. square space. Multiple different maze configurations are possible; an example map is depicted in Fig. 4. The course was constructed using medium-density fiberboard (MDF), 2×4 lumber and standard door hinges. The course consists of 1 foot high walls, 4 inch high 2 ft. by 2 ft. platforms, 2ft. by 2 ft. ramps leading to and from the platforms, and a speed bump made from layered pieces of MDF. Example of each building block of the maze can be seen in Fig. 5.

IV. RESULTS

Student performance is judged based on their ability to grasp robotics concepts and implement these concepts in the aforementioned laboratory exercises. We gauged that a student successfully mastered the concepts introduced in that lab experiment if they completed the exercise and lab report and earned 90% of the points for that assignment. While it may appear that this is a high level of achievement, we determined that, because of the "binary" nature of completion (it either worked or it did not), those who completed the assignment

TABLE 1. Data of success of mastering DaNI lab assignment concepts.

Offering (students)	2011 (n=12)		2012 (n=22)		2013 (n=31)		2014 (n=31)		2015 (n=46)		2016 (n=53)	
Lab Assignment	% Meet	Avg.	% Meet	Avg.	% Meet	Avg.	% Meet	Avg.	% Meet	Avg.	% Meet	Avg.
1. Introduction	83%	94%	100%	100%	75%	90%	94%	96%	93%	95%	87%	96%
2. Motor Control	100%	99%	100%	99%	71%	90%	81%	92%	100%	94%	94%	96%
3.Edge Detection	83%	95%	91%	97%	84%	94%	100%	98%	100%	100%	87%	87%
4. A*Path Planning	NA	NA	82%	94%	61%	93%	100%	97%	100%	100%	98%	89%
5. Obstacle Course	100%	98%	100%	100%	52%	92%	87%	97%	100%	98%	91%	91%

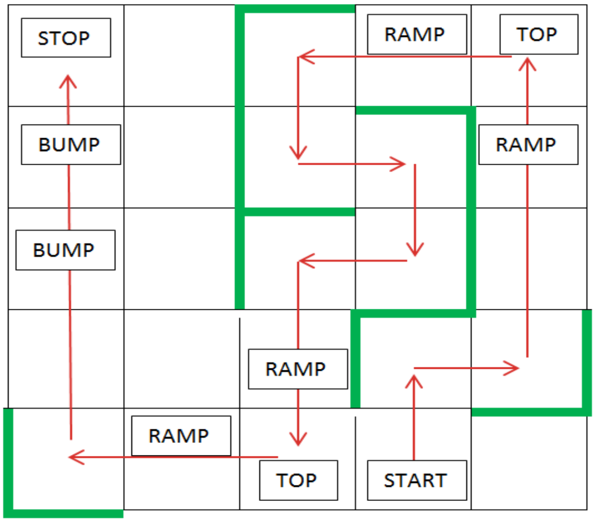

FIGURE 4. Map of DaNI obstacle course.

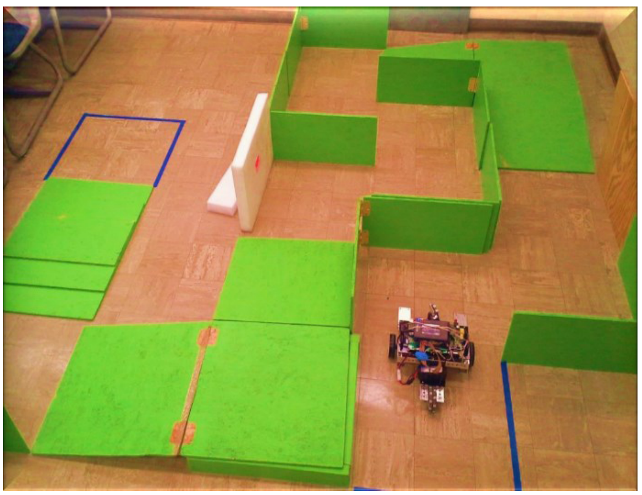

FIGURE 5. Bird's eye view of DaNI obstacle course.

typically earned all of the points, while those who did not typically earned below 90% of the points. The observed performance of undergraduate and graduate students was comparable.

Table 1 shows student performance of the five lab experiments over the last six years. The data shows very good success of the demonstration of their robotics concept knowledge with the DaNI robot task implementation. Each year of our data is broken out into two columns. One column (% Meet) shows the percentage of students in the class that earned 90% of that lab's points. The other column (Avg.) shows the average numbers of points earned by all students on that lab assignment.

The data shows that nearly all students grasped the robotics concepts, and in many cases 100% of the students demonstrated their mastery of the lab topics. The one semester with the poorest performance was 2013, where a combination of a new teaching assistant (sometimes not able to answer lab questions effectively) and the instructor's illness in the last 25% of the semester (not available to answer lab questions or cover course material live) led to lower scores, especially for the last two lab assignments.

Anecdotal evidence of student success and interest has been gathered from informal mid-semester and more formal end-of-semester evaluations. Through written comments student have told us that the lab experiences have been a highlight of the class and that the class has been the most enjoyable they have experienced at the University. Other anecdotal evidence of the value of this instruction is related to the number of students who have taken the course and later went on to jobs with robotics companies.

V. CONCLUSIONS

Several junior, senior, and graduate-level courses at UNC Charlotte have been taught which concentrate on electrical, computer, and mechanical topics related to autonomous control of robotic vehicles. Initially the courses used self-designed and built robotic platforms, but these robots caused students to spend too much time on subsystem activities like circuit assembly, circuit debugging, and device driver programming. A more robust commercial platform was needed in order to cover more interesting topics like component integration, sensor fusion, localization, mapping, and navigation.

The university purchased National Instruments DaNI robotic vehicles for use in robotics courses, which were augmented with additional sensors. Five laboratory exercises were developed that teach students such concepts like sensing, motor control, and path planning. Students concentrated on solving strategic tasks rather than lower-level hardware and software design. They were not however given canned solutions, as is done on other platforms; they had to do the electrical and programmatic integration of the sensor and actuator hardware.

The data for six years shows that nearly all students grasped the robotics concepts and demonstrated their mastery of the lab topics. The lab exercises and robotics class have proven to be a highlight of many students' college experience. Due to this success, the use of a pre-assembled industrial-based platform, with robust industrial level sensors and motors will continue to be used and developed.

VI. FUTURE WORK

As the students continue to experience success with the platform, they ask to be able to be able to do more with it. We have plans to continue to update the robots to allow them to be used with a variety of current and developing technologies, allowing the students to continue to build their portfolio of autonomous approaches. One such update will be to operate the DaNI robots using the popular environment ROS (Robot Operating System) [15].

We plan to add color IP cameras to the platform to make use of LabVIEW's vast library of image acquisition and processing libraries to give students the experience of processing a visual environment, and making logic decisions based on that feedback fused with other onboard data. This will allow us to do projects like having one robot track and follow another in a convoy like fashion.

With the recent development of low cost pseudo-LIDAR systems, we also plan to investigate adding these sensors to the platform. This would allow us to teach students how to map a 360 degree environment and fuse that with onboard sensor data to do more efficient path planning.

REFERENCES

[1] I. R. Nourbakhsh, E. Hamner, K. Crowley, and K. Wilkinson, "Formal measures of learning in a secondary school mobile robotics course," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 2. New Orleans, LA, USA, Apr./May 2004, pp. 1831–1836.

[2] I. R. Nourbakhsh *et al.*, "The robotic autonomy mobile robotics course: *Robot design, curriculum design and educational assessment*," *Auto. Robots*, vol. 18, no. 1, pp. 103–127, 2005.

[3] J. M. Gómez-de-Gabriel, A. Mandow, J. Fernández-Lozano, and A. J. García-Cerezo, "Using LEGO NXT mobile robots with LabVIEW for undergraduate courses on mechatronics," *IEEE Trans. Edu*, vol. 54, no. 1, pp. 41–47, Feb. 2011.

[4] M. Calnon, C. M. Gifford, and A. Agah, "Robotics competitions in the classroom: Enriching graduate-level education in computer science and engineering," *Global J. Eng. Edu.*, vol. 14, no. 1, pp. 6–13, 2012.

[5] N. A. Bousaba, J. M. Conrad, C. M. Hargrove, and V. Cecchi, "Keys to Success in the IEEE Hardware Competition," in *Proc. Annu. Conf. Expo.*, Vancouver, BC, Canada, Jun. 2011, pp. 22.990.1–22.990.18.

[6] J. McLurkin, J. Rykowski, M. John, Q. Kaseman, and A. J. Lynch, "Using multi-robot systems for engineering education: Teaching and outreach with large numbers of an advanced, low-cost robot," *IEEE Trans. Edu.*, vol. 56, no. 1, pp. 24–33, Feb. 2013.

[7] W. W. Walter and T. G. Southerton, "Teaching Robotics by Building Autonomous Mobile Robots Using the Arduino," in *Proc. ASEE Annu. Conf.*, Indianapolis, IN, USA, Jun. 2014, pp. 24.1170.1–24.1170.16.

[8] H. Weinert and D. Pensky, "Mobile robotics in education and student engineering competitions," in *Proc. AFRICON*, Sep. 2011, pp. 1–5.

[9] R. Dhaouadi and M. Sleiman, "Development of a modular mobile robot platform: Applications in motion-control education," *IEEE Ind. Electron. Mag.*, vol. 5, no. 4, pp. 35–45, Dec. 2011.

[10] T. Crenshaw, "Using robots and contract learning to teach cyber-physical systems to undergraduates," *IEEE Trans. Edu.*, vol. 56, no. 1, pp. 116–120, Feb. 2013.

[11] A. F. Browne, C. Benfield, and M. Calvin, "Enhancing mechatronic education with a low-cost conversion of a first generation DaNI robot to a second generation platform," in *Proc. ASEE Southeast Section Conf.*, Apr. 2015. [Online]. Available: http://se.asee.org/proceedings/ASEE2015/ASEE2015SE%20frame.htm and http://se.asee.org/proceedings/ASEE2015/papers2015/96.pdf

[12] T. Bower, "Teaching introductory robotics programming: Learning to program with national instruments' LabVIEW," *IEEE Robot. Autom. Mag.*, vol. 23, no. 2, pp. 67–73, Jun. 2016.

[13] *NI DaNI Robot Labs*. Accessed: Jun. 7, 2011. [Online]. Available: https://goo.gl/5Jb6UE

[14] R. Siegwart, I. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, 2nd ed. Cambridge, MA, USA: MIT Press, 2011.

[15] B. B. Rhoades, J. P. Sabo, and J. M. Conrad, "Enabling a national instruments DaNI 2.0 robotic development platform for the robot operating system," in *Proc. IEEE SoutheastCon*, Charlotte, NC, USA, Mar. 2017, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/document/7925293/

AIDAN F. BROWNE received the B.S. degree in physics and mathematics from Vanderbilt University, Nashville, TN, USA, in 2008, and the M.S. degree in biological engineering, the M.S. degree in electrical engineering, and the Ph.D. in biomedical engineering, from the University of Connecticut, Storrs, CT, USA, in 1995, 1997, and 1998, respectively.

From 1998 to 2001, he was an Electrical Systems Engineer with Hamilton Sundstrand Space Systems International Inc. From 2001 to 2004, he was an Electrical Systems Engineer with Hamilton Sundstrand Control Systems. From 2004 to 2010, he was a Lead Systems Engineer with General Dynamics Armament and Technical Products Inc. He is currently an Assistant Professor with the Department of Engineering Technology and Construction Management, University of North Carolina at Charlotte. His research interests include robotics, mechatronics, instrumentation, and sensing.

Dr. Browne currently serves as the Secretary of the IEEE Charlotte Section.

JAMES M. CONRAD (S'84–M'92–SM'98) received the B.S. degree in computer science from the University of Illinois at Urbana–Champaign, Urbana, in 1984, and the M.S. and Ph.D. degrees in computer engineering from North Carolina State University in 1987 and 1992, respectively.

From 1984 to 1990, he was a Software Engineer with IBM, from 1992 to 1995, he was an Assistant Professor with the University of Arkansas, and from 1997 to 2003, he was a Software Engineer and Project Manager with Ericsson and Sony Ericsson. Since 2003, he has been an Associate Professor and Professor with the Electrical and Computer Engineering Department with the University of North Carolina at Charlotte. He is the author of eight books and over 140 articles in the areas of embedded systems, robotics, parallel processing, and engineering education.

Dr. Conrad was elected to serve on the IEEE Board of Directors as Region 3 Director/Delegate from 2016 to 2017. He has been recognized by the IEEE Region 3 and the IEEE-USA with several awards for exemplary teaching and service.