Curricular Lessons Learned in Robotics Education: a decade in review

Prof. Joel M. Esposito, United States Naval Academy

Joel Esposito is a Professor in the Robotics and Control Engineering Department and the inaugural Director of Innovation at the United States Naval Academy. He teaches courses in Robotics, Unmanned Vehicles, Computer Vision and most recently undergraduate research methods. He is the recipient of the 2010 USNA Rauoff Award for Excellence in Engineering Education, the 2015 Class of 1951 Faculty Research Excellence Award and the Navy's Civilian Meritorious Service Medal. He is the former editor-in-chief of ASEE's Computers in Education Journal; and a trained ABET Program Evaluator. He is the author of over 75 peer reviewed articles, has appeared as a guest on NPR, and served on advisory panels for NSF, ONR, DoT, NASA, Office of the Secretary of Defense and the Marine Corps Intelligence Agency.

Dr. Jenelle Armstrong Piepmeier, United States Naval Academy

Dr. Jenelle Piepmeier earned a Bachelor of Science in Mechanical Engineering from LeTourneau University. She went on to earn a Master of Science and Ph.D. from Georgia Institute of Technology, also in Mechanical Engineering. For over 20 years she has taught robotics, computer vision, and control systems to the future leaders of the Navy and Marine Corps at the United States Naval Academy in Annapolis, Maryland. The 2014 recipient of the Raouf Faculty Award for Excellence in Teaching Engineering award, Jenelle currently serves as the chair of the Weapons, Robotics, and Control Engineering Department. Her research interests included vision-based robotic control at both the micro and macro scales. She is thankful that her father, a computer science professor, brought his ”portable” computer home and taught his young daughter how to write a for loop.

Lessons-Learned in Robotics Education:

a Decade in Review

Disclaimer: The views expressed in this document are those of the authors and do not reflect the policy or position of the U.S. Naval Academy, Department of the Navy, the Department of Defense, or the U.S. Government.

Introduction

As the oldest of the 6 undergraduate Robotics Engineering degree programs in the United States, we reflect on national trends and program-level lessons learned since we modernized our curriculum a decade ago. After a brief overview of our program, we discuss changes in the robotics education landscape over the last ten years, including the proliferation of degree programs, issues in accreditation, challenges in hiring, the expectations of students and administrative challenges. Some of the content is based on our own program observations and assessments, other data comes from a recent on-line survey sent to over 10,000 roboticists – a follow-up to the 2015 version found in [1].

Overview of the US Naval Academy's Robotics and Control Engineering Program

The US Naval Academy is the United States Navy's official undergraduate-only college. In addition to military, character and physical training, all students complete a STEM-based set of core courses in addition to coursework from their chosen major which begins in sophomore year. Upon graduation they are awarded a BS in one of 32 majors and commission into the Navy or Marine Corps as officers.

Our ABET-accredited program originated in the 1970s with a long-standing emphasis on mechatronics and feedback control. Robotics coursework was added in the 1990s, with offerings in mobile robotics, computer vision, autonomous vehicles and artificial intelligence added in the following decades. In the academic year 2013 -14 we modernized our by-design curriculum and in 2015 the name of the department was officially changed to Robotics and Control Engineering. The program is housed in a stand-alone department and graduates about 75 students per year, including our honors program.

The curriculum is centered around the project-based-learning (PBL) paradigm (Fig. 1)– with nearly every course including hands-on lab experiences and an open-ended final project. The curriculum is structured as follows.

- Sophomore Year: Programming (Fall) and Mechatronics (Spring) gateway courses that provide the foundation for future PBL experiences.
- Junior Year: A two-course sequence in classical and state space feedback control, one in modeling and simulation, and a mini-capstone design experience.

- Senior Year: A required course in articulated-style robotics plus four elective slots that embrace the breadth of modern robotics (e.g. embedded-systems, mobile robotics, advanced control systems, autonomous vehicles, computer vision, artificial intelligence and desktop manufacturing). All students complete an open-ended capstone design sequence, working in teams on projects of their choosing.

Figure 1: Students at USNA engaged in project-based learning.

Trend: An Increase of Undergraduate Robotics Engineering Degree Programs

Traditionally, faculty who worked in robotics held appointments in more traditional departments such as Mechanical Engineering, Electrical Engineering or Computer Science; and the students who studied under them received degrees in those disciplines with the word “robotics” nowhere to be found on their diploma. But after Worcester Polytechnic Institute launched the first BS in Robotics Engineering in the United States in 2007 [2], there has been a steady rise in the number of by-design programs.

The 2013 Roadmap for U.S. Robotics [3] set a goal of having 10 accredited bachelor programs by 2027. In 2015 a survey entitled “The State of Robotics Education” [1] identified 7 undergraduate degree programs. By 2023 [4] estimated there were 36 such programs just in the US. At the time of this writing (2025) we have confirmed there are at least 41 bachelor's-level engineering or engineering technology programs with the word “robotics” in their title across the globe. This remarkable growth – 41 new programs in 17 years – reflects both a maturity in the field of robotics as well as a trend in higher education toward niche degree programs. As these programs evolve, and even more come online, we present several challenges for consideration.

Challenge: Technological Change is Constant

A decade ago, a survey of 67 robotics educators [1], suggested MATLAB had a firm foothold in the classroom (61%) buoyed by its popularity in the adjacent computer vision and feedback control communities, but it had no official robotics toolbox. C and C++ were considered the *de facto* choice for any serious hardware developer (52%). ROS had a growing following (28%), as did Open CV (17%), Gazebo (11%) and to a lesser extent Point Cloud Library (3%) – all of which were written in C++, adding to the case for teaching that language. But then in late 2014, Willow Garage – the company that supported all of those projects – announced it would be dissolved. Webots (5%) and CoppeliaSim (formerly V-REP, 6%) were simulators competing with Gazebo.

Consider the conundrum faced by robotics educators at that time – trying to predict which of these tools would be considered an “in-demand” skill in the years to come. It would have been reasonable to decide to teach C++ as a foundational language and to resist the temptation to overly invest in the open-source packages given Willow’s dissolution. Or to guess that for-profit MATLAB would be abandoned in favor of free open-source efforts. It would not have been unreasonable to predict that software giant Microsoft might leverage its dominant position in the operating system market to catapult its Robotics Developer Studio to market dominance.

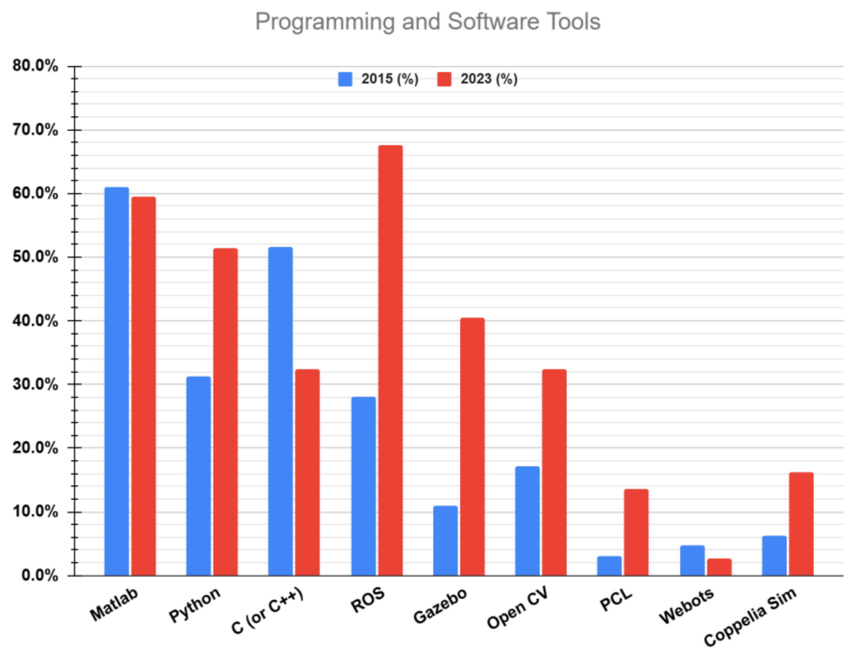

Figure 2: Programming environments used in robotics education: 2014 vs 2024.

Now a decade later our survey suggests the open-source packages have thrived outside of the stewardship of Willow Garage. ROS (68% -- up by 40 percentage points!) is now dominant. Open-CV (32%), and PCL (14%) have experienced a similar rise. Gazebo (40%) is the dominant simulator (it is built into ROS), though CoppeliaSim (16%) and WebBots (2%), continue to exist. Meanwhile Microsoft canceled its MS Robotics Developer Studio project, as part of its 2015 restructuring plan. We also see that the relative adoption of Python vs C / C++ has essentially flip-flopped, with Python growing from 31% to 51% and C/C++ shrinking from 52% to 32%.

When asked why they choose ROS for their classes, over half of the comments refer to it being a standard in industry and an expectation for their graduates rather than citing any superior technical feature of the language. As we shape our programs, one of our responsibilities is to prepare students for the workforce by arming them with in-demand skills and tools. While this is laudable, history suggests trends should be pursued cautiously, as it can be very difficult to predict what topics should be considered “fundamental” and what will be “in-demand” 5 years from now.

Challenge: An Evolving Accreditation Landscape

The Accreditation Board for Engineering and Technology (ABET) is the world’s leading accreditation body for technical degree programs. To attain accreditation, a program first selects a commission through which to apply (e.g. Engineering) and demonstrates compliance with that commission’s *general criteria*. In addition, programs must comply with any applicable discipline-specific program criteria—as indicated by the title of the program.

For the last decade, there was no program-specific criteria for “robotics engineering” so similarly titled programs only had to comply with the general criteria for engineering. However, ABET recently launched a “Robotics, Mechatronics and Similarly Named Programs” criteria– new for the 2024-25 accreditation cycle–stipulating that the curriculum include calculus, differential equations, linear algebra, and calculus-based physics; and that it covers engineering topics including mechanical systems, electronic circuits, control systems, and computer science [5]. The new criteria implicitly define robotics as the integration of hardware (sensors, actuators and embedded controllers) and software to control mechanical systems – rejecting software-only or mechanical-design focused approaches.

As of October 2024, there are at least 12 robotics and 38 mechatronic, previously accredited, engineering programs that will be subject to these new criteria at their next review cycle (sometime in the next 6 years); along with any new programs whose name includes the words “mechatronics” or “robotics”. It is advisable to review one’s curriculum, to ensure coverage of all of the topics listed. For example, a robotics program that is an outgrowth of a mechanical engineering department may need to add an embedded computing lab, while those with more of a computer science bent should be sure the curriculum does not omit feedback control systems or differential equations.

Challenge: Embrace the Breadth of Robotics

A recent surge in the robotics industry (and the adjacent AI boom), has put increased pressure on faculty hiring – both because undergraduate enrollment is increasing but also because PhDs have increasingly attractive job opportunities outside of academia. Filling faculty positions requires hiring committees to embrace the breadth of our field when it comes to the academic background of candidates.

For decades, a robotics course typically followed an outline from J.J. Craig’s or R.P. Paul’s classic texts from the 1980 – a kinematics-based analysis of serial chain articulated manipulators. By the 1990s ground-based indoor mobile robotics courses were added. In 2015 only 21% of professors described their course as “a broad and balanced introduction to robotics” [1] while in the most recent survey that number has increased to 49%. A look at the session tiles of any of the major robotics conferences reveals a wide variety of descriptors including: marine, aerial, medical, agricultural, networked, soft, micro and nano, wearable, vision-based, etc. -robotics. This suggests roboticists may come from a wide variety of academic backgrounds and that hiring ad language may be overly restrictive (e.g. "successful candidates will have a PhD in mechanical or electrical engineering").

Indeed, our own hiring data reveal a candidate pool with terminal degrees in biology, mechanical engineering, industrial engineering, electrical and computer engineering, engineering management, and computer science. We also note that a decade ago, our department would receive over sixty applicants. In recent years, our pool of applicants is closer to a dozen.

Challenge: Project-Based Learning

Project-based learning has proven to be a powerful tool for education - indeed 92% of the respondents to the survey employ hardware-based lab experiences and 72% a final project. Open-ended problems have the potential to excite and attract students, facilitate retention of material, promote lifelong learning, and encourage them to take ownership of their education. However, this paradigm also poses significant challenges including resources, and striking a balance between mass appeal and pedagogical rigor.

21% of faculty complain of a lack of space and departmental funds to host these experiences. In [1] the respondents in our survey frequently cited the dearth of affordable (< $10,000 USD) turn-key articulated robot manipulators for education - with 60% calling it one of their primary challenges. The situation is slightly less dire in the area of mobile robotics with the iRobot Create and Turtlebot available for under $1,000 (USD) however adding sensor packages can quickly increase costs. Kits such as Lego, Vex and Lynxmotion present further options but are not always appropriate for upper-level course work.

In response to this gap in the market offerings, a large percentage of programs seem to be relying on internally developed hardware (and software [6]) setups but we are wary of this approach. While it requires a smaller cash outlay, it comes with hidden costs. Significant faculty and support staff labor is required to maintain such systems ($60-$200 per hour including overhead). While this cost may not be billed directly it comes at the expense of other activities. And home-grown setups often come at the expense of the student experience as they are less reliable and user-friendly.

Finding the correct balance between lecture and lab is another challenge of problem-based learning. In our Junior-level two-credit guided design class (4 hours of lab, no scheduled lecture), the common design project integrates all the elements of a feedback control system from modeling, controller design, sensor calibration, etc. Nominally, most of the topics and skills have been covered in a previous course. Yet instructors work to balance a just-in-time instruction to facilitate a productive lab period while providing room for trial-and-error experiences. In polling the students about time spent to complete course deliverables, we found that on average, they were spending in excess of three hours outside of class (exceeding the Carnegie credit expectations of 2 hours). As a result, we are adding a lecture hour to this course, to give faculty the opportunity to provide just-in time instruction while still giving students the longer lab period to implement and troubleshoot.

Challenge: Gateway Experiences

Designing a gateway curriculum for incoming students presents the pedagogical challenge of bringing in students with a wide range of experiences with robotics. Many students have no exposure at all. When asked "What is the biggest challenge in robotics education?" [1] the most commonly cited issue (47%) was teaching students with such a wide range of academic backgrounds, skill sets and interests." Some have been exposed to robotics competitions, summer camps, or STEM experiences where robotics was used as a medium for teaching students a wide range of mechatronic, computing, or mathematical skill sets. When asked "Do you feel the academic preparation of your students is deficient in any of these areas?", 57% of the respondents answered computer programming.

In the fall of 2024, we asked our introductory students taking their first programming course in our major the following. *If you participated in a high school robotics club, class, or competition, how did that prepare you for this course? What surprised you about your experience in this course?* While roughly half of our students have some sort of prior experience in coding or robotics competitions, the other half have none. Students with prior experience said it provided them with a "basic familiarization with … terms", made "me more comfortable with coding" and "homework a lot easier." However, some students with prior experience express surprise that they weren't controlling "actual robots" (we used LEDs and DC motors). Even those with prior experience expressed surprise at the complexity of coding at the undergraduate level.

Gateway experiences do not have to be limited to programming. University of Michigan's curricular map starts with their innovative linear algebra course [7], *ROB 101: Computational Linear Algebra* which facilitates the ability to tackle computational robotics problems without the gatekeeper of three semesters of calculus and differential equations. With linear algebra less common in high school offerings, the course provides a more even playing field.

Challenge: Institutional and Organizational Considerations

The interdisciplinary nature of robotics draws faculty from a variety of disciplines, and robotics programs have evolved out of a number of engineering and computing departments, with mechanical engineering, electrical engineering, and computer science owning ~64% (Figure 3). While there is a broad tent for robotics educators to provide inspirational experiences and foundational knowledge, we feel that single department ownership of a robotics program is critical to success. As Jenkins observed in [8], until he joined Michigan Robotics, his answer to the best major for a roboticist was a "quadruple major in Mechanical Engineering, Electrical Engineering, Computer Science, and Mathematics … with a minor in Psychology!" Single department ownership prevents faculty silos and fractured budgetary resources.

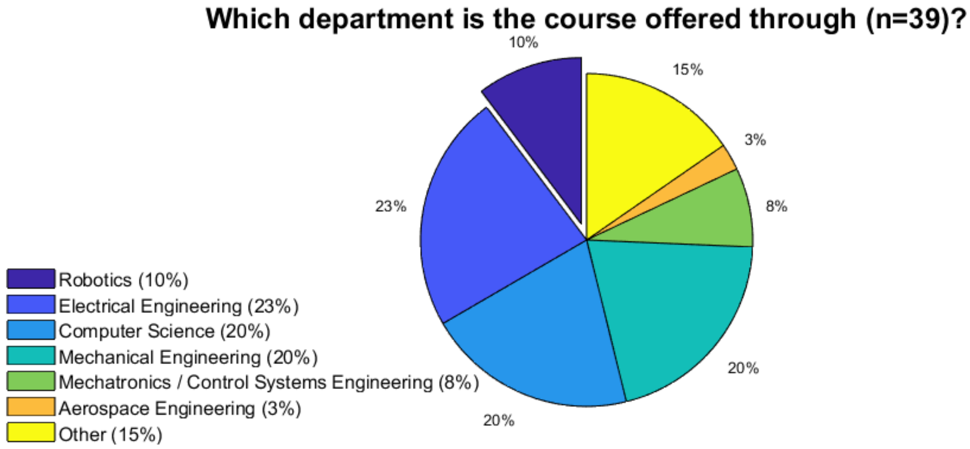

Figure 3: Survey response data on host departments for robotics courses.

As undergraduate robotics programs emerge in the higher education landscape, it can be insightful to look back at the history of higher education for the past century and a half. In the post-civil-war era, the United States found itself in the midst of the emergence of new institutions and models of pedagogy fueled by the industrial revolution. *The Great Upheaval* [9] contrasts the educational reforms at Harvard led by Charles Eliot with the University of Chicago, constructed by William Rainey Harper. Eliot brought about an "incremental layering of reforms," and Harper had the opportunity to "construct the American university from scratch." Eliot's effort is likened to building "an airplane from a horse cart while flying it." For the authors, we have watched our own program evolve from a feedback control systems-focused program with a couple of robotics electives to a much broader selection of upper-level elective robotics and advanced control offerings. We have been a part of an evolutionary, incremental growth versus the ground-up programmatic design at Michigan and WPI [2, 7, 8].

Several efforts [10,11] have surveyed the robotics community to assess the most common topics covered in robotics programs helping to reveal common curricular threads. In [9] the authors note that the standardization of academic programs during this wave of educational reform that continued into the early part of the 20th century was brought about in part by the accreditation organizations, such as the Engineers' Council for Professional Development (ECPD) - ABET's precursor- which first began accrediting engineering programs in 1932. ABET's addition of program criteria for "Mechatronics, Robotics, and Similarly Named Engineering Programs" may be a first step in bringing about some standardization in an undergraduate robotics curriculum.

When polled, there is not a broad consensus about the need for standardization of programs. Of the respondents, ~25% disagree or strongly disagree on the need for standardization, while ~49% agree or strongly agree on the need for standardization. What stands out, as shown in Figure 4, is the move from a neutral position on the topic towards an opinion, one way or the other.

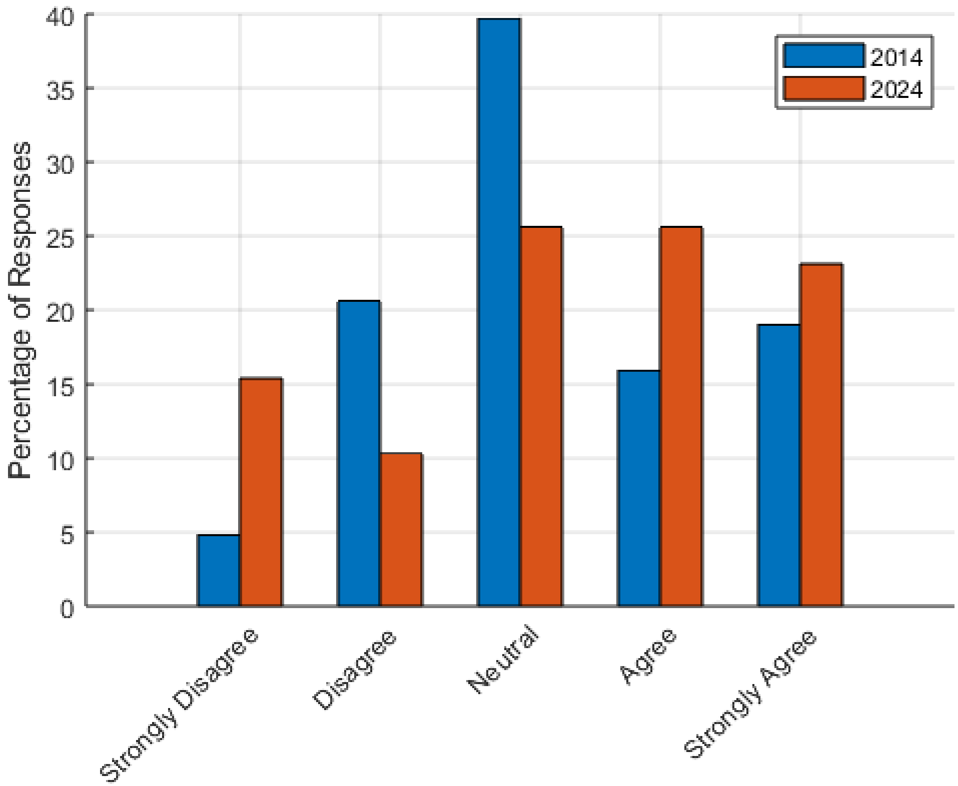

Figure 4: Likert scale responses for creating a standardized curriculum in robotics.

When asked about the future of robotics education, several respondents suspect AI may replace kinematics and dynamics. Others suggest an increased role of virtual and augmented reality for demonstration and explanation of three-dimensional systems. Topically they see a growth in service and field robotics, and less emphasis on industrial robotics. In terms of instructional methods, a growth in online teaching materials and simulated robotics labs is predicted by one respondent.

Conclusion

This article presented data from a survey of robotic educators – a follow-up to a similar survey conducted a decade ago [1]– combined with anecdotal and assessment evidence from our home department to reflect on trends and challenges in robotics education. We noted explosive growth in the number of by-name Robotic Engineering degree programs (41 new ones since 2007); and described the newly created ABET accreditation criteria these programs will need to comply with. We reflected on the changing technology landscape (especially in programming languages); the challenges of project-based learning and the need to balance fundamentals with the latest in-demand skill sets; and the need for considering various academic disciplines when hiring. Finally, we discussed organizational considerations – advocating for single department ownership when possible.

In the *Great Upheaval* [9], LeVine and Van Pelt suggest that new content delivery methods and demographic pressures will disruptively change the higher academic landscape. Looking forward, robotics educators face similar challenges of disruptive technologies (e.g. AI/ML) and educational models (such as competency-based learning, and any-time, any-place education).

References

[1] J. Esposito, "The State of Robotics Education: Proposed Goals for Positively Transforming Robotics Education at Postsecondary Institutions," *IEEE Robotics & Automation Magazine,* vol. 24, no. 3, pp. 157-164, Sept. 2017.

[2] M. Gennert and C. Putnam, "Robotics as an Undergraduate Major: 10 Years' Experience," in *ASEE Annual Conference & Exposition*, Salt Lake City, Utah, 2018.

[3] "A Roadmap for U.S. Robotics From Internet to Robotics 2013 Edition," sponsored by Robotics in the United States of America, Robotics VO, March 20, 2013.

[4] S. McLeod, "An Analysis of Undergraduate Robotics Degree Programs in the United States of America," *IEEE Robotics and Automation Magazine.* In press.

[5] [Online]. Available: https://www.abet.org/accreditation/accreditation-criteria-for-accrediting-engineering-programs-2024-2025. [Accessed 1 November 2024].

[6] J. Esposito, O. Barton and J. Kohler, "MATLAB toolbox for the iRobot Create," 2008. [Online]. Available: https://www.usna.edu/Users/weaprcon/esposito/_files/roomba.matlab/ .

[7] P. Apte, "(SOLVING FOR) EQUITY," *ASEE Prism*, vol. 30, no. 4, pp. 34-37, 2021.

[8] O. Jenkins, J. Grizzle, E. Atkins, L. Stirling, E. Rouse, M. Guzdial, D. Provost, K. Mann and J. Millunchick, "The Michigan Robotics Undergraduate Curriculum: Defining the Discipline of Robotics for Equity and Excellence," *arXiv.2308.06905,* August 2023.

[9] D. Levine and D. VanPelt, *The Great Upheaval*, Johns Hopkins University Press, 2021.

[10] C. Berry, M. Gennert and R. Reck, "Practical Skills for Students in Mechatronics and Robotics Education," in *ASEE's Annual Conference and Exposition*, 2020.

[11] K. McFall, K. Huang, H. Gilbert, M. Jouaneh, H. Bai and D. Auslander, "Mechatronics and Robotics Education: Standardizing Foundational Key Concepts," in *ASEE Annual Conference and Exposition*, 2020.