2023 Annual Conference & Exposition
Baltimore Convention Center, MD | June 25 - 28, 2023
The Harbor of Engineering
Education for 130 Years
ASEE
Paper ID #40083

Development of a Cobot Lab to Support Next-Generation Applied Engineering Technology

Jeritt Williams, Illinois State University

Development of an Industrial Cobot Lab to Support Next-Generation Applied Engineering Technologists

Introduction

One of the big ideas of the Industry 4.0 concept is that modern manufacturing is shifting from mass production to customized production, signaling the need to deploy advanced technologies that allow the future workforce to work in a flexible, highly productive, and adaptable way [1]. To this end, collaborative robots (cobots) have been identified as an essential connection to the goals of Industry 4.0. They are considered "…one of the leading intelligent factory innovations that make the global production market competitive and productive" [2]. The use of cobots in manufacturing-related activities has rapidly increased over the past several years and shows signs of significant growth potential [3]. Recent research suggests that future manufacturing-related roles, such as those often filled by applied engineering technologists (e.g., machinists and manufacturing engineers), will benefit from the knowledge and use of cobots to support their work [4]. These are just a few reasons to suggest that applied engineering technology educators consider how they might utilize cobots in preparing future workforce-ready graduates.

Engineering Technology faculty at Illinois State University redeveloped an existing Integrated Manufacturing Laboratory (IML) to include five industrial cobots to be used concurrently with five six-axis articulated industrial robots in an undergraduate-level, applications-focused robotics systems integration course. This paper describes the rationale for deploying industrial cobots into a traditional industrial robotics systems integration course. It describes the lab redevelopment process, provides initial assumptions and early observations, and discusses lessons learned to date. The next steps for research and practice are also outlined.

Background

The IML was initially established in 2007 and was designed to serve up to 20 students working in pairs. Ten identical workstations were designed to include an ABB IRB140 six-axis articulated industrial robot (IRB), an Allen-Bradley PLC, Human Machine Interface (HMI) display, a networked PC, and various industrial components and peripherals (Fig. 1). When plans were made to update the IML with equipment and resources to support the "next generation" of engineering technologists, faculty attended trade shows and spoke with industry advisors before ultimately determining that including industrial cobots in the IML was the direction to pursue. Five of the original IRBs were replaced with ABB CRB15000 "GoFa" (CRB) cobots (Fig. 2).

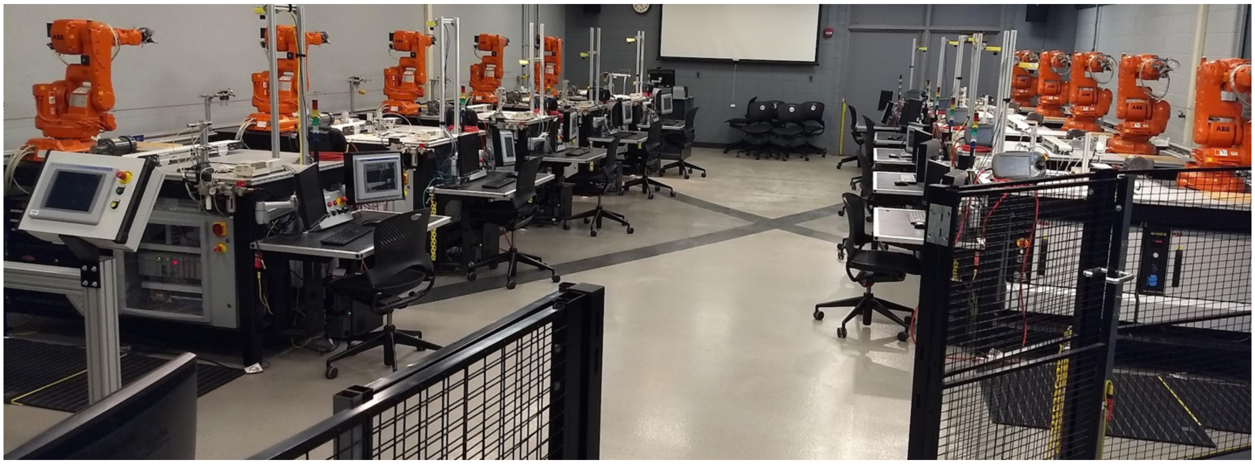


Fig. 1. Original IML layout.

Fig. 2. New IML layout. (Left) Five IRBs with original workstation layout. (Right) Five CRBs with the redeveloped design.

Discussion

The current objectives of the undergraduate curriculum are to introduce students to basic robot operations, working safely with robots, programming fundamentals, use of external I/O, and creating applications predominately associated with material handling. The following describes ongoing developmental considerations and observations from lessons learned during the pilot semester.

Laboratory Layout and Setup

The decision to maintain the overall aesthetics of the lab was a top priority for the faculty for various practical reasons. Five of the original IRB stations would remain in place, and five stations would need to be remodeled, removing the IRB to feature the CRB. Each IRB station was initially built using custom worktables designed to efficiently hold everything required for the station and provide a flexible space for lab activities. The initial renovation plan was to replace the IRB with the CRB, leaving the remaining workstation as-is. However, while the CRB's work envelope and payload are similar to the IRB, it was determined that important ergonomic factors had been overlooked. The CRB relies on tactile action buttons for jogging and control at the robot's "head" (Fig. 3). This would likely cause most users difficulty in controlling the robot during activities such as lead-through teaching and jogging, as the necessary controls would be out of reach and view of the operator. To address this issue, new, commercially available tables were purchased and modified to lower the robot to an ergonomically appropriate position while maintaining the use of the custom-made aluminum extrusion work surfaces. This allowed faculty to achieve their goal of the continued usage of all previously utilized peripherals and components. Other aspects of the “standard workstation” were also implemented in the CRB station design. For example, the main power disconnect switch from the removed IRB stations was integrated into the new CRB stations so that students followed the same startup and shutdown procedures at each station.

Additionally, the teach pendant was mounted similarly on all stations, and the function keys were set up to control the gripper in a standard format. All robots contained the standard “calibration routine,” which placed the robot in a home position. These decisions were made deliberately to assist students in transitioning more easily between the IRB and CRB systems, which they would alternate use for each class period. Faculty observed that these design choices were adequate.

Mobile Cart CRB Workstation

Fig. 3. Buttons located at “head” of CRB.

Fig. 4. CRB mobile cart in machine tending application.

The inherent flexibility and purpose of cobots preser hentic learning experiences, mainly if the CRB station is free to move to another space, like the machine shop, for example, where it could be used in applications like machine tending or as an additive manufacturing center (Fig. 4). Furthermore, the faculty agreed that a mobile CRB station could be helpful in marketing and recruitment. It could be moved relatively quickly and used for interactive demonstrations, like those at trade shows. However, it was found during the re-installation of the lab that accommodating a mobile station had many unique challenges, especially in the integration of the PLC and connected peripherals. Careful planning, retrofitting, and rewiring were often required to accomplish the task. To address this issue, the mobile CRB workstation was split into two parts: A rolling cart to carry the CRB and a separate table for all other elements (Fig. 5). Many cables were modified with harnessed connectors that allowed for quick field connections.

Fig. 5. Mobile CRB in “split” rolling workstation.

Safety

The built-in safety design of the CRB has shown to be an improvement in both the deployment of the equipment and its use in the curriculum. The CRBs included *SafeMove*, which integrates many safety features into settings that can be demonstrated visually in the offline programming software. This allowed faculty to provide supplemental instruction about safety topics to include real examples of safety concepts and configuration without altering the physical robot, allowing the lab to remain in “production mode” for other courses that may require using the same equipment. Additionally, the CRB’s reduced operational speed and integrated sensors ensure that it stops and pauses operation if an operator or object bumps it. This presumably reduces the risk of potential injury or damage to the surrounding workstation. In early observations, students have been more productive in their initial use of CRB versus their initial use of IRB. One potential reason is that they can resume and correct their errors instead of waiting for the instructor to utilize an administrator-level system login to fix the issue before returning control to the student operator.

A significant difference between the CRB and IRB is that the CRB can be programmed in automatic mode and is designed to run automatically while the operator is in the shared workspace. This allowed for a reduction in the multiple redundancies in the existing safety guarding system. Since the CRB was designed to promote human-robot interactions in a shared workspace, several current safety guarding elements were reduced or removed (See Fig.1, 2, and 5). This included the removal of floor mats and safety chains (gates) that blocked the work cell from operating in automatic mode when an operator was present in the work cell, a 3-position key switch that is required to engage automatic mode on the IRB systems, and a reduction to the sophisticated programming of a safety PLC.

During the installation of the CRB stations, it was found that end-of-arm-tooling (e.g., grippers) was much more limited in terms of commercially available products than their IRB counterparts. To overcome this challenge, it was determined that suitable EOAT could potentially be created by 3d printing. However, special considerations should be taken to comply with guidelines set forth by relevant robotics safety organizations, such as ensuring that sharp points are blunted, and pinch points are minimized or eliminated [5] and that risk assessment is performed for every job (similar to the use of IRB).

Basic Operations

The goal of the curriculum was to continue teaching the fundamental basic operations pertaining to the operation of industrial robots, including jogging, online teaching of programs, creation of tool data, and use of inputs and outputs (I/O) using each of the robot types.

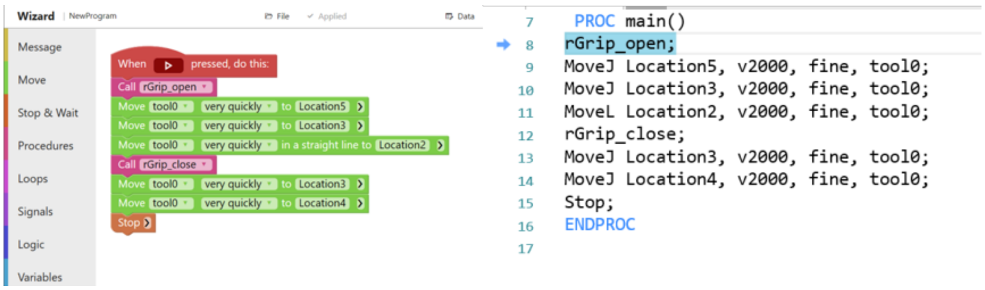
As the course got underway, the most divergent concept between the IRB and CRB was the lead-through jogging and teaching available on the CRB. The CRB can be programmed in lead-through in either manual or automatic modes, both of which allow the operator to control the movement of the robot by physically grabbing joints of the CRB and pushing or pulling them into a position before setting the position using the action button located at the head of the robot (Fig. 3) or by saving the coordinate on the teach pendant. When practicing jogging and teaching positions using this method, students who worked on the CRB completed their activities noticeably faster than those on IRB stations.

Tool data creation is often tricky for students. When controlled using the IRB or manual jogging mode on CRB, it requires careful control over setting the coordinate system on the teach pendant and finesse on the joystick. It was assumed that students would be significantly more efficient using the CRB. Early observations indicate that this task is comparatively challenging even when using lead-through techniques on the CRB.

The decision was made to use a commercially available electric gripper on the CRBs. This presented a challenge in uniformity because the IRBs used a pneumatic gripper. The main difficulties of using a different tool were that the finger dimensions were not uniform between the IRB and CRB stations, and the I/O signals used to communicate with the tool differed. Alternatives of the objects used in lab activities, such as material handling, were created and implemented. Initially, the faculty were concerned with making the CRB activities match those of the IRBs as closely as possible. As the lab activities were underway, it became evident that this was less important than was first thought. The CRBs offered new opportunities for expanding student engagement, such as designing custom tooling using methods like 3d printing and advanced simulation using online programming software, especially since they could complete the physical lab activities quicker on the CRB than those working on similar activities on the IRB stations.

Programming

The CRB allows students to use *Wizard*, a Blockly-based form of drag-and-drop programming that automatically generates the underlying RAPID (text-based program). In concept, this will enable students to create a fully functioning program without the prerequisite knowledge of the traditional coding method (see example in Fig. 6).



```
When ▶ pressed, do this:
Call rGrip_open
Move tool0 very quickly to Location5
Move tool0 very quickly to Location3
Move tool0 very quickly in a straight line to Location2
Call rGrip_close
Move tool0 very quickly to Location3
Move tool0 very quickly to Location4
Stop
```

```
 PROC main()
rGrip_open;
MoveJ Location5, v2000, fine, tool0;
MoveJ Location3, v2000, fine, tool0;
MoveL Location2, v2000, fine, tool0;
rGrip_close;
MoveJ Location3, v2000, fine, tool0;
MoveJ Location4, v2000, fine, tool0;
Stop;
ENDPROC
```

Fig. 6. Wizard (left) vs. RAPID (right) programming comparison.

Early observations suggest that the time students develop their working applications has improved using Wizard to generate the corresponding RAPID code. However, these IRB stations do not currently utilize Wizard programming, so students will likely need to become proficient in RAPID and less reliant on Wizard in the long term if they are to be experienced in both systems. Some students have indicated that Wizard has helped them understand how RAPID instructions are composed, thus allowing them to transition away from Wizard. They have also stated that once they are comfortable in RAPID (and using the corresponding development software, RobotStudio), they are less likely to continue using Wizard as they begin to find the text-based coding approach to be faster and more efficient. An introductory programming course with a non-robotics focus is required before taking the robotics course to give students an essential background in programming. Further investigation is needed to determine if block-based programming provides a sufficient basis for the introduction and knowledge transfer required to work successfully in text-based coding in RAPID.

Systems Integration

Integration with existing systems (e.g., PLCs, HMI, industrial network devices, etc.) has been a primary concern for faculty working on this project. There were initial challenges in retrofitting the CRBs with some existing systems. This was often due to the age of existing systems and technological advances since their installation. The faculty of this project sought to keep existing peripherals and components in place as often as possible for various reasons. A primary concern was that industrial peripheral components are often costly, and replacing these components might not make sense if they are otherwise fully functional. When possible, newer versions of the technology were added to replace previous systems. For example, machine vision cameras were upgraded to work better with the embedded capabilities of the robot programming software. The effectiveness of this upgrade has yet to be determined but is planned for future phases of this project.

Projects

The possibility and breadth of new activities supporting hands-on, project-based learning are still under investigation. The initial concern of faculty was whether previous projects and lab activities used with IRB stations would work as-is with CRBs to continue the original goals of teaching the fundamentals of industrial robotics systems. It was quickly realized that this was not necessarily the case, nor was it likely a high priority. While the CRBs may offer the potential for incorporating many new activities and explorations of new and alternative concepts, they must also be juxtaposed with the realities of time, resources, and other academic constraints.

Next Steps for Research and Practice

In continuing to seek to support the development of next-generation engineering technologists, utilizing cobots to engage with other critical aspects of Industry 4.0 is a worthy choice. There are opportunities to work with digital twins and simulation, additive manufacturing, and AI/ machine learning. The robotics software used with IRBs and CRBs offers a chance to bring industry 4.0 concepts such as digital twins and simulation into the curriculum [6]. While the current curriculum includes digital twins of the IRB and CRB stations, where students create and test applications offline before deploying them to the physical robot, recent technological advances have allowed for incorporating extended and mixed realities with offline programming tools to aid in planning, programming, and visualizing operations [7]. The next steps of the lab development will be to revisit the curricular goals and opportunities of the program to determine if this focus can be added to the existing course or if there is sufficient need to warrant future courses and capstone experiences.

CRBs may offer a unique opportunity to explore other next-generation manufacturing activities, such as engaging in additive manufacturing like 3-d printing or providing a solution for human-robot interactive welding. Digital twining and simulation will be integral aspects of these processes as well. Integrated machine vision capabilities will likely add new dimensions of learning and research, especially in teaching concepts like machine learning [2].

Finally, there is an opportunity for researchers to investigate how the programming of the CRB impacts productivity, particularly as it relates to block-based programming on the CRB vs. traditional text-based coding on the IRB. Recent research has shown potential in this area [8], [9]. Future work in this area may also consider how AI can assist in coding and application development.

Conclusion

While curriculum and laboratory development conclusions are preliminary, a few things have become apparent to the faculty stakeholders. Teaching and learning in an environment that utilizes both IRB and CRB stations concurrently shares more in common than the faculty initially assumed. Faculty were originally concerned about how they could make the lab activities as similar as possible to help students navigate the basics of learning the robotic systems. However, this may be a minor concern, as the CRBs have proven intuitive, mainly when used concurrently with the IRBs. Students often finished tasks on the CRB in less time than IRB when given the same task. Students are learning more in less time. This has opened space in the curriculum and created an opportunity for student-faculty research in previously tricky areas, given the time, resource, and safety constraints of the IRB-only space.

References

[1] S. Vaidya, P. Ambad, and S. Bhosle, “Industry 4.0 – A glimpse,” *Procedia Manufacturing*, vol. 20, pp. 233–238, Jan. 2018, doi: 10.1016/j.promfg.2018.02.034.

[2] M. Javaid, A. Haleem, R. P. Singh, and R. Suman, “Substantial capabilities of robotics in enhancing industry 4.0 implementation,” *Cognitive Robotics*, vol. 1, pp. 58–75, Jan. 2021, doi: 10.1016/j.cogr.2021.06.001.

[3] J. Kite-Powell, “Collaborative robots make a comeback In 2021,” *Forbes*. https://www.forbes.com/sites/jenniferhicks/2022/04/19/collaborate-robots-make-a-comeback-in-2021/ (accessed Feb. 12, 2023).

[4] E. Peiffer, F. Plaza, S. Shamsi, E. Collins, and A. Huderson, “Future of mechanical engineering, manufacturing engineering, and machinist roles for industry 4.0,” presented at the 2022 ASEE Annual Conference & Exposition, Aug. 2022. Accessed: Feb. 11, 2023. [Online]. Available: https://peer.asee.org/future-of-mechanical-engineering-manufacturing-engineering-and-machinist-roles-for-industry-4-0

[5] Robotic Industries Association, “Technical report for industrial robots and robot systems - collaborative robots,” ANSI, Technical Report RIA TR R15.606-2016, Dec. 2016.

[6] J. Williams and K. L. Devine, “Developing a digital twin for highly flexible learning modalities,” in *Proceedings of the 2021 ATMAE Conference*, Orlando, FL, Nov. 2021, pp. 67–76.

[7] A. Rivera-Pinto and J. Kildal, “Visuo-tactile mixed reality for offline cobot programming,” in *Companion of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, in HRI ’20. New York, NY, USA: Association for Computing Machinery, Mar. 2020, pp. 403–405. doi: 10.1145/3371382.3378290.

[8] D. Weintrop, D. C. Shepherd, P. Francis, and D. Franklin, “Blockly goes to work: Block-based programming for industrial robots,” in *2017 IEEE Blocks and Beyond Workshop (B B)*, Oct. 2017, pp. 29–36. doi: 10.1109/BLOCKS.2017.8120406.

[9] D. Weintrop *et al.*, “Evaluating CoBlox: A comparative study of robotics programming environments for adult novices,” in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, Montreal QC Canada: ACM, Apr. 2018, pp. 1–12. doi: 10.1145/3173574.3173940.