



Collaborative Human-Robot Assembly: Methodology, Simulation and Industrial Validation

Junshen Chen, Abdullah Mohammed^(✉), Theocharis Alexopoulos,
and Rossitza Setchi

School of Engineering, Cardiff University, Cardiff CF24 3AA, UK
alhusinalkhdurag@cardiff.ac.uk

Abstract. This paper developed an approach to integrate a human-robot collaborative assembly with a manual production line. The integration was completed effectively by exploring several collaboration algorithms. The study was conducted to enhance the working environment by improving the productivity and the quality of the production. Variability in the assembly processes was reduced, thus improving product quality and reducing rejects and reworks. The automation or robot collaboration was evaluated on one of the existing assembly lines, to help worker reduce repetitive work and increase productivity, which will also help to save labor costs over the long term. The inspection outputs from a robot are easily accessible, providing the quantitative data, analysis of which will lead to continual improvement.

Keywords: Collaborative robot · Industrial application · Robot assembly

1 Introduction

Robots have blended into humans' lives for several decades to assist humans in a large number of application areas. Recent years have seen new developments in human–robot co-assembly operations across the manufacturing sector. Global competitors and technological advancements have resulted in the use of such systems in more challenging and complicated manufacturing environments [1]. At the same time, a hybrid assembly task that combines a robot-assisted system with the human worker [2] has numerous vital and outstanding advantages, which substantially lessen the amount of fixed production costs in comparison to variable costs. Collaborative robotics triggered a huge shift from the traditional robot-in-a-cage model to robots interacting with people in a fenceless environment [3]. Consequently, human–robot collaboration is getting increasing attention in manufacturing applications. The cooperation of humans and robots in collaborative assembly tasks can take advantage of the differing strengths from both sides [4]. Generally, the robot provides several benefits including reduced worker fatigue and increased productivity. Human-robot collaboration therefore results in reduced levels of physical strains for human partners [5] while the human partner provides an incomparable ability

to operate high accuracy sensory components, and rapidly adapt to new and complex subtasks.

The conventional coding approaches for the collaborative robots are gradually becoming less able to satisfy the flexibility and variability of product developments, especially those with more personalized and customized requirements [6]. As a result, more efficient and rapid programming approaches need to be developed to enable the robot to be quickly adapted for new tasks in human–robot collaboration.

Programming by demonstration (PbD) is a similar technique to the record and replay technique, in which a robot is shown a set of movements and then repeats them exactly multiple times, but PbD has an aspect of learning integrated into it, making it a more effective system. In contrast to conventional coding procedure, PbD allows this process to be streamlined by showing the robot its task, while its position, joint rotations and any other required pieces of data are recorded. This allows it to repeat the task by following this data and no coding is required. It is also possible for the robot to learn how to deal with varying circumstances by showing it through multiple different but similar scenarios, enabling the robot to generalize its task. For example, Cousins et al. (2017) demonstrated the possibility of replicating the movement of the user's hands with robot's hands, allowing the robot to be adjusted remotely [7]. Wang et al. (2018) developed a teaching-learning-collaboration (TLC) model for the collaborative robot to learn from human demonstrations and assist its human partner in shared working situations [8].

During the last decade, the emergence of asset digitalization has enhanced the development of robot processes with simulation tools that can accurately represent the human-robot setup [9]. The work of Pieska et al. (2018) tests a number of modern robot process development tools where a simulation environment is used together with robot teaching or high-level programming to program the robot offline. Then the high-level code is translated to a low-level code that the robots understand and execute [10].

Cobots are inherently passive robots whose goal is to facilitate a full collaboration between the human and the robot. This collaboration means that the human and the robot will share the working envelope, the task and the process. This work presents the method followed to introduce cobots in the assembly process of a brake booster. The aim is to develop and validate an environment of a human-robot collaborative assembly. In such an environment the human and robot will work in cooperative mode where they will share the working space and the time. In more detail, 1) Introduce a system using cobots to assemble parts at production rates equal to or higher than the current ones. 2) Increase worker productivity during the assembly of two pistons, two extensions and two plungers. 3) Reduce repetitive work undertaken by workers 4) Minimize the effort to reconfigure the system for different parts 5) Design the system taking health and safety standards into consideration. 6) Achieve the results without the need for part tracing/identification.

The developed system is explained in the second section of this paper followed by the third section which describes the methods used in the approach. The fourth section illustrates the research results, and the final section presents the conclusions.

2 Methodology

To determine the type of human-robot collaboration, the distribution of tasks and the sequence of the distributed tasks follow a three-stages process. Firstly, the manual tasks are analyzed. Then, a simulation model that includes both the human and the robot is developed. The aim of the simulation model is to optimize the independent and collaborative activities as well as to ensure that the ergonomics of the assembly cell allows for the execution of all tasks and does not impose increased or unnecessary strain to the human. Finally, the tasks are individually validated at the purpose-built test bed with the actual robot and the equipment and tools required for the assembly. More detail on each of these stages is provided in the next sub-sections.

2.1 Assembly Procedure Analysis

In this section, every step of the assembly procedure is analyzed to decide where to deploy the robot to handle certain tasks. The following is the details of the criteria that have been used:

- The complexity of achieving the task by the robot: This aspect is needed to evaluate the challenges that the system integrator can face during the installation and programming of the robotic system. The complexity at this point is analyzed from the robot perspective. This means that the nature of the task will play a major role in deciding the level of complexity as some tasks are easier to be performed by the human and vice versa.
- The complexity of robot control: This criterion analyses two aspects in the robot: the first one is the kinematics of the robotic arm, the second one is the low-level control of the robot's joints. This is needed to control the robot movement.
- Reliability of the robot deployment: This criterion is measured by looking at the output of the assembly and evaluate the quality for several cycles. This will give indication about the level of trust in implementing the robotic solution in the manual production line.
- Possibility to deploy the robot using the current system and equipment: This criterion analyses the status of the production line the identify the benefits of implementing the robot in the existing setup. This helps the company to evaluate the feasibility of using the robot from the environmental, ergonomic and economical perspectives.

A summary of human-robot co-assembly procedure analysis is given in Table 1. The subassembly breakdown is categorized into seven main operations. Green color is the prior operations to implement the robot. Orange colored operations need further investigation due to the constrains of using current assembly tools. Red operations are easier for human operators due to the time limit of this project.

2.2 Production Cell Simulation

A novelty of this research project is that after the assembly process analysis a hybrid development methodology is followed that combines simulation in parallel to actual robot testing in a purpose-built testbed. Core objectives of the simulation model are:

Table 1. Assembly procedure analysis

Categorised operations	Human	Robot	Constrains with current system and equipment
Pick-up parts, Place to nest	Repetitive work. Requires pre-unload parts for convenience of latter operations	Reliable. Adding the robot can save time and efforts	Grasping point of each individual part needs to be identify
Place bullet over part	Need to switch between different sizes of bullet for different parts	Achievable. In this step, the robot will do a repetitive job with one program	Design extension to the current bullet for robot to use these current tools
Pick up O-rings/seals from container	This step can be done by a human operator to eliminate the robot control complexity	Achievable but complex with limited project time	May require advanced sorting machine
Pick up pusher, push O-ring/seals down into groove	Speed depends on the skill of workers	Achievable. In this step, the robot will do a repetitive job with one program	Design extension to the current pusher for robot to use these current tools
Pick up circlip	This step can be done by a human operator to eliminate the robot control complexity	Challenging for robots to pick up single item from jumbled tray	
Sizer parts	Repetitive work	Achievable. Can be repeated/extended to different size without extra efforts	The grasping point of the part need to be identified
Store parts	Repetitive work. Can be done without extra effort if last step is done by human	Reliable. Adding the robot can save time and efforts	The grasping point of the part need to be identified

- Assist in the development of the control logic,
- Provide a robot cell visualisation to ensure that the available hardware can be integrated and that both human and robot tasks can be done efficiently
- Enable an iterative process where simulation results are embedded into the testbed and testbed results are embedded into the simulation model.

Due to the nature of the simulation required, Siemens Process Simulate software package was selected. This simulation package supports a detailed human and robot simulation as well as virtual robot controllers that can be programmed like the actual controllers allowing for code sharing and validation in both virtual and physical environments.

The first step for the simulation model development was to create a number of different hardware setups that were proposed after brainstorming. These models contained all hardware and tools that were either available or possible to build without any interactions considered since this was the base to define the system logic. Some of the project participants used CAD software to express their ideas so the parts could be easily imported into the simulation software package. Visualization is important to enable the engagement of non-technical people that can provide valuable insight about the goals of the change in the production.

After a generic setup of tools, equipment and human working space was defined, the system interactions (logic) were modelled. This step took into account the assembly procedure analysis in order to separate the tasks that are done by the robot and establish sequences and relations or checks between all tasks (human and robot).

Then an iterative process was initiated where a specific process setup and logic was modelled and the tasks whose result was not deterministic were flagged to be tested on a physical system. A characteristic example are the tasks that involved the pushing of washers or O-rings into the respective grooves. The simulation model cannot ensure that the end-result would comply with the quality standards and in addition new robotic tools were developed to achieve the desired result. In these cases, the task was developed and tested on the physical testbed and the results were then passed to the simulation model in the form of a 3D CAD model of the new tool when new tools were developed and in the form of robot programming code when a different task execution was required.

Finally, the simulation model was used to make the production cell more ergonomic in aspects such as easy reach of parts and tools by the human, minimization of the collaborative space to improve human safety, and development of an easy to follow routine for the human to reduce the required levels of focus. The simulation model cannot be developed without the testbed (described in Sect. 3.3), and it is not intended to provide an independent offline solution that would be downloaded to a production cell robot only at the end of the development process (as it would be done in idealized process development cases).

2.3 Task Validation Testbed

As explained in Sect. 3.2, not all tasks could be designed through the simulation model. Installing washers and O-rings accurately required a redesign of the tool (pusher) that pushed the washer/O-ring to the correct position. Then due to the bespoke nature of the solution, details such as pushing angles/forces/speeds, and optimum tool gripping position had to be defined through a trial and error process. To develop these tasks a modular testbed was built where all trials could run. Since cobots can be easily relocated the testbed was installed at a controlled space outside the production to ensure the safety of personnel working on other tasks and prevent obstructions in production that shopfloor trials may cause.

The testbed consists of a table where the robot or other equipment can be mounted on (at any position), the robot with its gripper, manual part conveyors and storage boxes or trays. The testbed can be easily reconfigured which is important especially in the initial stages of development where the orientation of equipment and even the flow direction of parts has not been decided.

Parallel development (simulation and testbed) requires that tasks are designed as modules with discrete input and output in both the physical and the digital worlds. Taking the fitting of a washer as an example, the task input is the positions and orientations of the robot, of the pusher, of the washer expansion bullet and of the washer itself before the task begins. The output is the positions and orientations at the end of the task. To simplify the development of the cell the robot joints have zero speed at the beginning and at the end of the task and the state of each model item remains the same (for example no changes in shape or temperature). In addition to the physical system parts the digital input is the active signal(s) before task starts and the digital output is the signals generated at the end of the task (as a result of the task). This modular approach facilitates the transition between the physical testbed and the simulation model and ultimately enables the development of the cell partially as a simulation model and partially as a physical model. In the washer fitting example, the simulation model was developed with a generic task representing the washer fitting. This task was then developed and executed at the testbed using the input from the simulation model. The results were measured (in terms of time, position and orientation) and finally transferred to the simulation model along with the robot program.

Overall, the testbed could be considered as an extension of the simulation model. The latter provides a more complete picture of the final production cell but it lacks the accurate measurements, the quality checks and reliability verification which can only be done at the testbed.

3 Case Study Demonstrations

3.1 System Description

The manual assembly cell produces master cylinders for automotive brakes which are twin boosted to reduce up to 90% of the required pedal effort. The parts that this work assembled with a human-robot collaboration are the pistons (0.165 kg), the extension pistons (0.120 kg) and the plungers (0.338 kg). Examples of the parts are shown in Fig. 1.

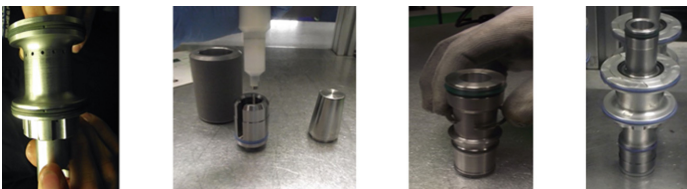


Fig. 1. From left to right: piston, piston extension, plunger, full assembly (piston tower)

Prior to assembling the parts, a number of seals and O-rings need to be mounted on the parts as well as a valve that is installed inside the piston extension. These form the biggest challenge either for the human that needs to follow precisely multiple steps or for the robot that must ensure that everything is mounted accurately.

The manual process that must be followed to prepare the piston, piston extension and the plunger is shown in Fig. 2. After the completion of Fig. 2 steps the three parts are ready to be installed in the cylinder. Since these are parts of a twin booster system a set of 2 pistons, 2 extensions and 2 plungers are needed for each booster assembly. Installation into the cylinders is done at a later stage and is not considered for the human-robot collaboration.

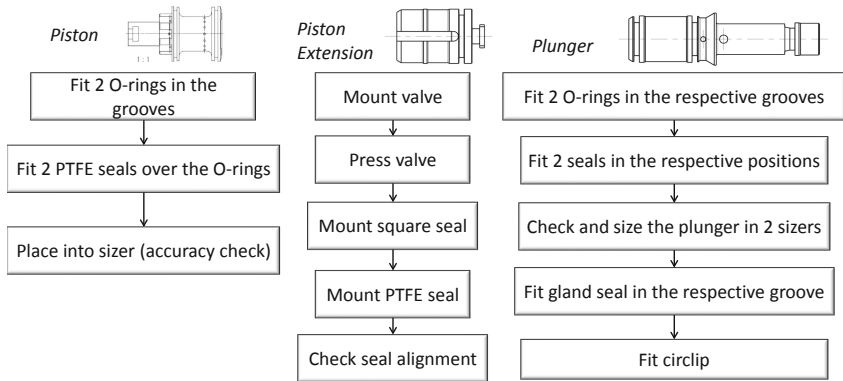


Fig. 2. Manual assembly steps

The average time needed for a skilled worker to assemble a piston, an extension piston and a plunger is 15 s, 40 s, and 45 s respectively. The total time for each twin booster is 200 s. Due to the nature of the manual work, there are numerous factors that can affect these times. However, the average was calculated by measuring the worker performance on one part and not on the average number of parts during a shift.

For the transformation of the assembly cell a Universal Robots (UR) UR5e robot [11] has been selected. UR robot is designed to make collaborative robot technology accessible to small and medium-sized businesses. UR5e is a 6-axis industrial manipulator with a weight of 20.6 kg, the maximal payload of 5 kg and a working radius of 850mm, featuring built-in torque/force sensors.

In manual control mode of the robot, the user can move each joint separately or move all joints synchronously by setting a goal position for the end-effector of the arm. A teaching mode is also available on the user's interface, which allows the user to move the arm by hand to the desired position.

The UR5e robot can run on automated control as well, and the programming can be performed at two levels, the Script Level and C-API Level. In case of Script Level programming, the arm is only controlled by a program written in URScript, a language developed by Universal Robots for the UR robot series. The URScript language is similar to Python programming language, and one can use variables, data types, functions and the flow of control statements. It also provides necessary commands for communication and motion control, but it does not include extended libraries for motion planning or advanced mathematics.

The grippers picked for this project are from SCHUNK, the Co-act EGP-C [12]. It is certified in accordance with ISO/TS 15066 and integrates easily and fast with the UR5e cobot.

3.2 Simulation Results

The first outcome of the simulation was the design of the human-robot collaboration cell. This is depicted in Fig. 3. All parts flow in the direction of the arrow shown in the figure. The human receives the trays containing the items to assemble, prepares the parts for the robot (which includes placing washers and O-rings over the respective expansion bullets) and operates the press that inserts a valve into the piston extension. Then the human places the parts in the respective designated areas and the robot finishes the assembly by pushing the washers and O-rings into the piston or plunger. Finally, the robot places the parts into a sizer and moves them to an area where the next worker can pick them up and continue the assembly process. All equipment and parts as well as the position of the human and the robot are placed in a way to minimize travelling distances. The human reaches assembly items and tools easily with minimum bending and the robot makes the most of non-collaborative areas where it can run at maximum speed and therefore reduce cycle times.

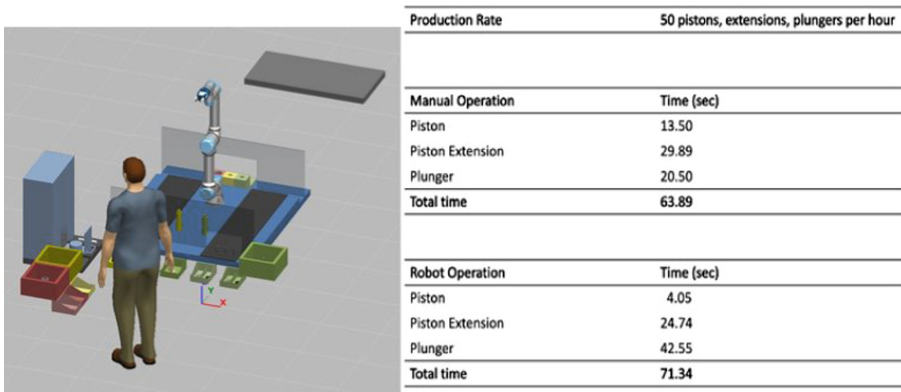


Fig. 3. Left: the simulated cell. Right: Simulation results

The second outcome of the simulation is the process logic. The robot operates in ‘opportunistic’ logic which selects between available tasks with a first in first out method. As soon as assembly items are available in the designated area the robot starts the task related to these parts. If parts are available in 2 different designated areas, then the robot carries on the task related to the parts that became available first.

The third outcome was the calculation of the collaborative process times. Figure 3 shows the summary of times for the groups of tasks carried out by the human (discussed in Sect. 4.2) and the robot. The robot is slower than the human and the overall cycle time equals the robot cycle time (72 s) which is 28% less than the cycle time of the manual process.

3.3 Testbed Results

The feasibility of the purposed robot-assisted assembly operations were tested and demonstrated on the testbed as shown in Fig. 4. Firstly, teaching by demonstration method was used in robot programming. This shortened the learning curve for the new operators and allowed them to use the graphical user interface (GUI) for programming the robot. This method also provides redundancy of manipulating other customized parts in different dimensions, which meets the needs of the company. This method was used throughout the development of the process to create the programs that were then passed to the simulation software.

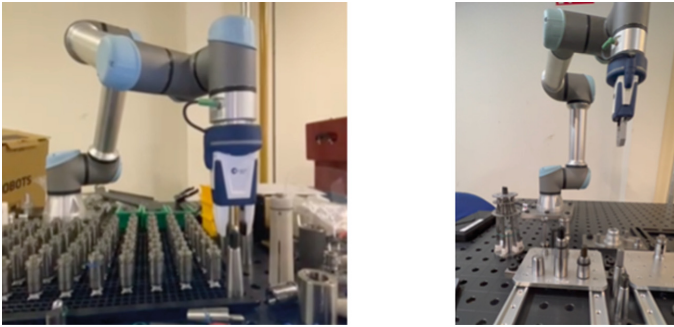


Fig. 4. Robot assembly tasks validation. Left: Piston extension. Right: Plunger

As mentioned in Sect. 3.3, a new type of pusher was designed to improve the efficiency of robot operation of installing O-rings and seals. Additional adapters handles and stands were also designed and used for robot gripper to interact with different tools. All these parts were tested in terms of finishing quality and process reliability.

4 Discussion and Conclusions

One of the main key findings that has been demonstrated in the reported research is the feasibility of introducing a collaborative robot to a manual assembly process. This is achieved by developing a hybrid model in which the robot and the shopfloor operator are working side by side. For the development, a novel approach is followed where a simulation environment and a physical testbed are running in parallel and provide feedback to each other. It is found that the introduction of the collaborative robot improved the productivity compared with the manual setup. Furthermore, the integration time for the collaborative robots was relatively short since they came with integrated safety sensors which limit the need for external safety considerations.

Two main challenges have been identified during the development of the approach. The first one is that the approach need to guarantee the safety of the operator working side by side with the robot. The second challenges are to find the balance between the speed of the operator compared with the robot as they need different times to perform the same task.

Acknowledgements. This work has been supported by the ASTUTE2020 (Advanced Sustainable Manufacturing Technologies) operation, which assists manufacturing companies across Wales, and has been part-funded by the European Regional Development Fund through the Welsh Government and the participating Higher Education Institutions.

References

1. Michalos, G., Makris, S., Papakostas, N., Mourtzis, D., Chryssolouris, G.: Automotive assembly technologies review: challenges and outlook for a flexible and adaptive approach. *CIRP J. Manuf. Sci. Technol.* **2**(2), 81–91 (2010)
2. Helms, E., Schraft, R. D., Hagele, M. rob@ work: Robot assistant in industrial environments. In: *Proceedings. 11th IEEE International Workshop on Robot and Human Interactive Communication*, pp. 399–404. IEEE (2002, September)
3. Wang, W., Li, R., Chen, Y., Jia, Y.: Human intention prediction in human-robot collaborative tasks. In: *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pp. 279–280 (2018, March)
4. Kru"ger, J., Bernhardt, R., Surdilovic, D., Spur, G. Intelligent assist systems for flexible assembly. *CIRP Annals* **55**(1), 29–32 (2006)
5. Surdilovic, D., Radojicic, J.: Robust control of interaction with haptic interfaces. In: *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 3237–3244. IEEE (2007, April)
6. Pires, J.N.: New challenges for industrial robotic cell programming. *Ind. Robot.: Int. J.* (2009)
7. Cousins, M., Yang, C., Chen, J., He, W., Ju, Z.: Development of a mixed reality based interface for human robot interaction. In: *2017 International Conference on Machine Learning and Cybernetics (ICMLC)*, Vol. 1, pp. 27–34. IEEE (2017, July)
8. Wang, W., Li, R., Chen, Y., Diekel, Z.M., Jia, Y.: Facilitating human-robot collaborative tasks by teaching-learning-collaboration from human demonstrations. *IEEE Trans. Autom. Sci. Eng.* **16**(2), 640–653 (2018)
9. Thomas, A., Guerra-Zubiaga, D.A., Cohran, J.: Digital factory: simulation enhancing production and engineering process. In: *ASME International Mechanical Engineering Congress and Exposition*, Vol. 52019. American Society of Mechanical Engineers (2018)
10. Pieskä, S., Kaarela, J., Mäkelä, J.: Simulation and programming experiences of collaborative robots for small-scale manufacturing. In: *2018 2nd International Symposium on Small-scale Intelligent Manufacturing Systems (SIMS)*. IEEE (2018)
11. UR5e robot. <https://www.universal-robots.com/products/ur5-robot/> Accessed 14/04/2022
12. Schunk Co-act EGP-C gripper. https://schunk.com/gb_en/gripping-systems/series/co-act-egp-c/ Accessed 14/04/2022