Development of A Manufacturing System for Gear Assembly using Collaborative Robots

Guoyuan Li, Erlend Holseker, Arvin Khodabandeh, Isak Gamnes Sneltvedt, Erik Bjørnøy and Houxiang Zhang
Norwegian University of Science and Technology
Postboks 1517, N-6025, Aalesund, Norway
{*guoyuan.li, erlehols, arvink, isakgs, eribjorn, hozh*}*@ntnu.no*

***Abstract*—Using robots in manufacturing processes is becoming more and more common. Robots can perform repetitive tasks that are not suitable for humans. Using robots in production can also save both time and money, as well as increasing product quality. However, using robots in complex assembly tasks can be challenging. Narrow spaces and tight fits can cause problems for the robots. Consequently, the paper presents a case study where two collaborative robots are used to assemble a planetary gear. In addition, the paper describes how a mobile robot is used to deliver the components used in the assembly. Image processing is used to orient the robots according to each other and localize the correct parts for each robot. One of the robots is also equipped with a force sensor which enables the ability to determine when a part is mounted correctly. The collaborative robots communicates through the Modbus/TCP protocol, while the mobile robot communicates through ARCL (Advanced Robotics Command Language) commands. All communication flows via a PLC, making the system a small-scale manufacturing system.**

***Index Terms*—Collaborative robots, mobile robots, gear assembly, industry 4.0**

I. Introduction

Industry 4.0 is considered the next industrial stage in which manufacturing process integration and product connectivity can help improve the industrial performance of companies [1]. Intelligent manufacturing is a concept of manufacturing within the Industry 4.0 philosophy which aims to optimize production and product transactions by using advanced information collected by different types of sensors [2]. Industrial Internet of Things (IIoT) as a central part of intelligent manufacturing and Industry 4.0, connects networked smart objects, generic information technologies, and optional cloud or edge computing platforms to improve production efficiency and product quality. This allows for the system to be real-time, intelligently, and autonomously accessible [3], [4], [5]. It also enables collection and analysis of data, communication between objects, and exchange of process and product information within the industrial environment, which improves overall production value.

In recent years, robots and intelligent systems have been becoming more and more widespread in variety of areas, such as path planning [6], advanced control [7], and handling complex assembly tasks [8]. According to statistics, assembly time accounts for 40% - 60% of the total manufacturing time in the automotive industry [9]. It is common that robots assist human workers throughout the whole assembly process, and some of the tasks can be done solely by robots [10]. Robots are well-suited for repetitive tasks, such as painting car bodies and welding, because they are very precise and consistent [11].

However, complex assembly tasks such as assembling an engine or a gearbox are not well-suited for robots because a majority of the parts in an engine are very small, and the space is often very narrow. In addition, car manufacturers often let the clients choose the specification of the car themselves, which means that there can be a great variation in the amount of equipment on a car. In such a context, it is hard for robots to mount the car bodies autonomously owing to the unique specification of the car.

Our research project aims to use the collaborative robots and mobile robots in Manulab to provide an autonomous manufacturing solution for complex assembling tasks in the industry. Manulab is a national infrastructure for basic research on manufacturing. The laboratories are open for all researchers and industry in Norway. The Manulab at NTNU in Ålesund is equipped with 3D-printers, collaborative robots, mobile robots, a laser cutter, a delta robot, and a cell consisting of two vertically articulated robots. The main purpose of the Manulab is to let industrial companies and researchers test different manufacturing solutions on a smaller scale or proof of concept using the equipment in the lab, without having to build a new factory to test the solution. In this paper, we propose an autonomous manufacturing solution that uses the collaborative robots and mobile robots in this lab for assembling gears in the industry.

The rest of the paper is organized as follows. Section II introduces an overview of the collaborative robots and mobile robots. In Section III, the key technologies used for assembling gears using collaborative robots are presented. Section IV presents an experiment of assembling a planetary gear. Finally, conclusion and future works are drawn in Section V.

II. Robots for assembling tasks

This section introduces the specifications of articulated collaborative robots and mobile robots that are used in the gear assembling tasks.

A. Collaborative robots

The collaborative robots used are Omron TM5-900, as shown in Fig. 1. These robots are 6-jointed, 6-DOF equipped with an integrated Eye-in-Hand camera. Table I is a list of the collaborative robots used in this project and the sensors

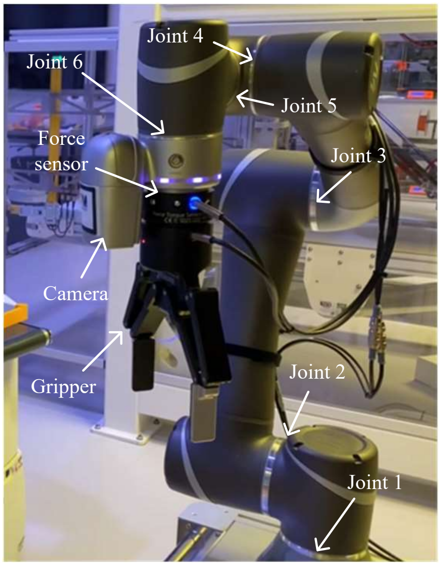


Fig. 1. Overview of a 6-jointed articulated collaborative robot.

Fig. 2. Mobile robot Omron LD-60.

TABLE I
COMPONENTS USED IN THE MANUFACTURING SYSTEM

Component	Amount	Description
Omron TM5-900	2	Collaborative robot with integrated camera
Robotiq FT 300	1	Force sensor
Robotiq 2F-140	1	Gripper, adaptive
Robotiq Hand-e	1	Gripper, parallel

and grippers they are equipped with. One of the robots are equipped with a force/torque sensor. The collaborative robots are used to assemble all the parts. This assembly requires communication and cooperation between the robots in order to be able to fit the parts together.

The robot arms have 6 motors each. Joint 1 and 6 has a movement range of $\pm 270°$, joint 2, 4 and 5 has a range of $\pm 180°$ and joint 3 has a range of $\pm 155°$. The reach of these particular robots are 900mm, and max payload is 4 kg [12]. The TM5 robots are programmed using TM Flow. TM Flow is a flow chart based programming environment. Within this software, every aspect of the robots can be controlled, including but not limited to movement, safety settings, communication and vision jobs [13].

The Eye-in-Hand camera on the robots is used to detect objects. The image processing framework are integrated in Omron's TM Flow software, and features different options such as morphology, contrast enhancement and pattern matching [14].

The force/torque sensor (F/T sensor) used is a Robotiq F/T 300. This sensor can measure force and torque in x, y and z. The force range is ± 300 N and the torque range is ± 30 N/m. The recommended force threshold is 1 N in x-, y- and z-axis, and torque is 0.02 N/m in x- and y-axis and 0.01 N/m in z-axis [15].

B. Mobile robot

The mobile robot used is an Omron LD-60, modified with a conveyor belt on top, as shown in Fig. 2. The robot was equipped with several sensors, including LiDAR, laser, force sensors and sonar. The robot is programmed using the software mobile planner [16].

With the use of LiDAR the mobile robot generates a map of it's locality. This is achieved by using the SLAM-algorithm [17]. Several force sensors are mounted throughout the robots body. This is for collision detection, and precision navigation. The laser and sonar sensors equipped is used for safe navigation, such as obstacle avoidance. The robots communicates with the ARCL standard [18], and is connected to a mainframe which is the hub of communication.

III. GEAR ASSEMBLY USING COLLABORATIVE ROBOTS

A. The overall procedure of gear assembly

For the gear assembly, two collaborative robots are used. The robots communicate and cooperate in order to achieve tasks that require "two hands" or a socket to hold the object still. In order to verify the assembly ability of the collaborative robots, a task of assembling a planetary gear set is chosen. Fig. 3 depicts an exploded view of the gear to be assembled.

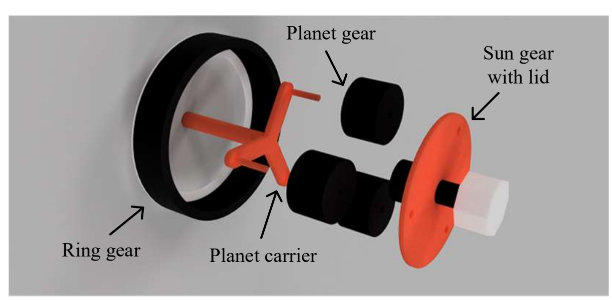


Fig. 3. Gear components for assembly.

It consists of one ring gear, one planet gear carrier, three planet gear, and one sun gear.

Fig. 4 illustrates the overall procedure of the assembly. The dashed arrows indicates communication between the robots while the solid arrows lead to the next step for each robot.

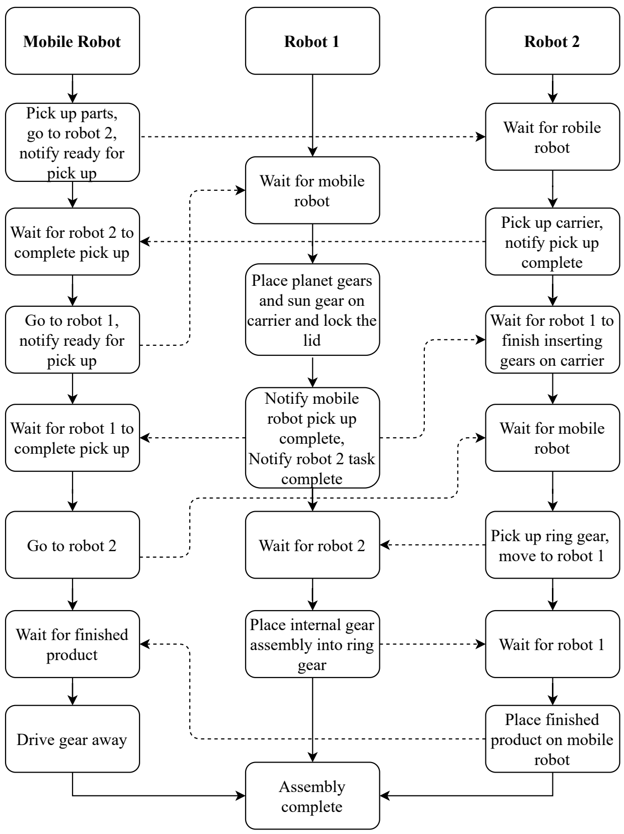


Fig. 4. Flowchart of the overall process in the assembly.

B. Using mobile robot for components delivery

The mobile robot's role in the assembly is for transportation of components. Firstly, it picks up the 3D printed components and then brings the different components to its respective drop off locations. As illustrated in Fig. 4, the mobile robot drives back and fourth between the robots in order to supply the robots with the different parts as the process proceeds. Lastly, the mobile robot drops off the finished assembled product at its designated drop off location.

The assembly process is initiated when the robot receives a command from the master node. The different pick up and drop off locations is saved in the robots inherent map. When the start command is received the robot navigates to the pickup location. Upon arrival the robot then goes idle, awaiting the next command. When the following command is received the robot will move with the unassembled parts from the start location to the area of assembly. After it has arrived a command is sent to the collaborative arms, that initiates the collection of parts, before it enters a idle state. This process is subsequently repeated until the robot has received the finished product. After the finished product is placed on the robot the last step is to transport the assembled gear to the drop off destination.

C. Using marker for location of grasping

The robots uses specific markings to orient themselves during the assembly. The markings are referred to as landmark. When one of the robots finds a landmark it compares the size and shape of the landmark to the image that was taken when the program was made. It then calculates its current orientation to the landmark. After the robots scan the landmark, the robot can use it as a base instead of the robot base. This ensures that the assembly would not fail even if the robots is not placed in an exact position.

At the beginning of every assembly, the robots start by localizing each other. One of the robots places itself in a certain position towards where the other robot is standing. The other robot will then try to localize the landmark which is placed on the Eye-in-Hand camera of the first robot. When the landmark is detected, the following points can be placed with this landmark as the base. This ensured that the robot always positions itself according to the other robot, even if the robot base is moved. The same method is also used when the robot is picking up the different parts. The robot takes an image of the parts and find the correct one in the image. A base is then made on the part (see Fig. 5) and the robot uses this base to place itself correctly according to the part prior to gripping it.

D. Recognizing parts placed on the mobile robot

All the parts are placed on customized trays before they are placed on the mobile robot. There are two different trays, each containing different sockets for the parts used by each of the collaborative robots. The trays are also equipped with a landmark and a QR-code. The QR-code and the sockets

Algorithm 1 Locating the trays and verifying correct tray

First: Wait for the mobile robot to arrive
while not *MobileRobotInPosition* **do**
Stay in while loop until mobile robot is in position
end while
Second: Move to the first tray and scan for landmark
Move to $FindTray1Pos$
while not *correctTrayFound* **do**
    if $foundLandmark$ **then**
        Move to $LocateQRPos$
        Scan QR code
        if $QRcode == "Robot2Tray"$ **then**
            $correctTrayFound == True$
        else
            Move to $FindTray2Pos$
        end if
    else
        Move to $FindTray2Pos$
    end if

Algorithm 2 Inserting planet gears using force sensor

while $F_z < 2.5N$ **do**
  Move down towards carrier
end while
while $F_{xy} < forceLimit$ **AND** $distanceTraveled < distanceLimit$ **do**
  Move in a spiral motion
end while
Release planet gear

are placed in a fixed position according to the landmark. This allows the collaborative robots to easily find the different parts on the tray. After scanning the landmark, each robot proceeds by scanning the QR-code in order to verify that it has found the correct tray. This sequence is described in Algorithm 1.

E. Using force sensor for guiding gear components

A force sensor is used to ensure that the components are assembled correctly. The force sensor is attached between the end of the robot arm and the gripper and gives a constant feedback to the program of the applied forces. The force sensor enables a series of new functions in the program which uses the input from the force sensor and compares the input to a predetermined limit. A distance limit is defined so that the robot can move while performing this action until it reaches the maximum distance. The mechanism is useful when inserting pins into holes. Algorithm 2 describes how the force sensor is used to insert the planet gears on the planet carrier.

F. Using gripping functions to avoid error

The "Grip" function has multiple output blocks including "detected", "not detected" and "error". The process continues on one of the output blocks depending on whether the robot manages to grip the object or not. This can be utilized to verify that an object is gripped, hence, avoiding a situation where the robot is suppose to grip an object and move it whereas in reality failed. Also, by programming different processes for the different outputs, actions can be repeated until the desired output is true. This mechanism is used when the robot locks the lid in place to ensure that the lid is locked correctly and the logic is described in Algorithm 3.

Algorithm 3 Locking the lid in place

Try to grip the lid
if $LidGripped$ **then**
  $LidGrippedSuccessfully = True$
else
  $LidGrippedSuccessfully = False$
end if
while $LidGrippedSuccessfully$ **do**
  while $T_{xy} < torqueLimit$ **AND** $distanceRotated < rotationLimit$ **do**
    rotateLid
  end while
  Try to grip lid
  if $LidNotGripped$ **then**
    $LidGrippedSuccessfully = False$
  end if
end while
Move down to grip lid
Grip lid
while $T_{xy} < torqueLimit$ **AND** $distanceRotated < rotationLimit$ **do**
  rotateLid
end while
Release lid

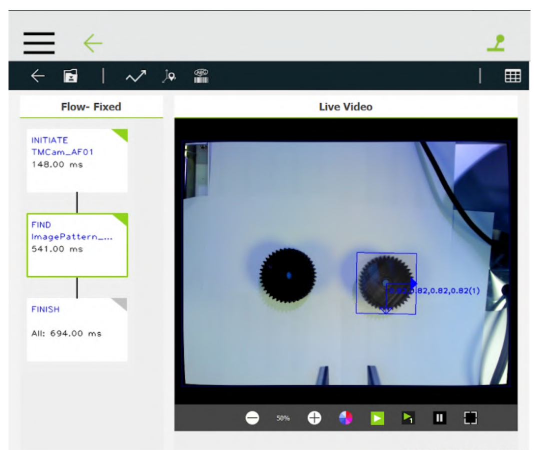
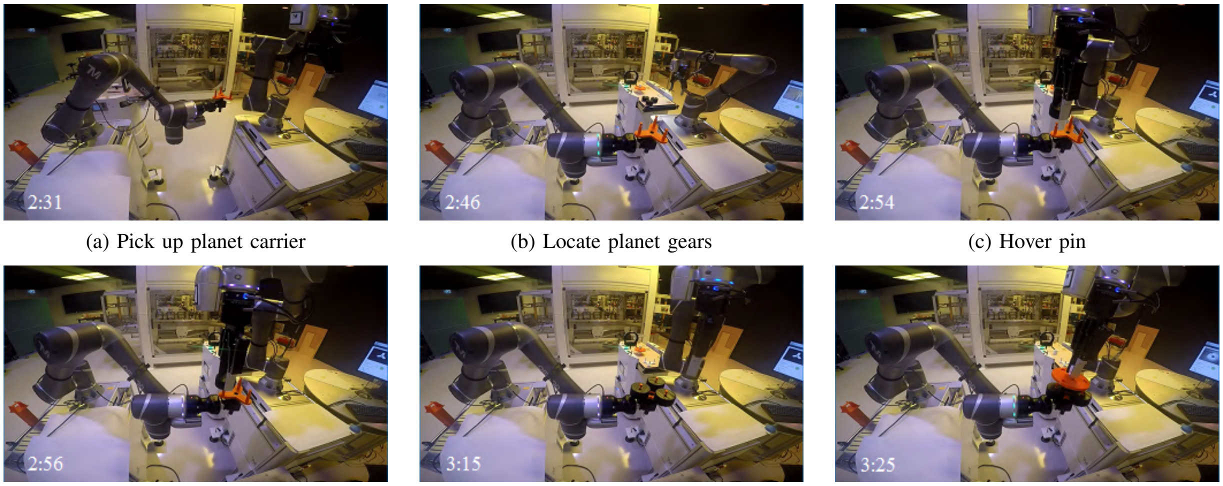


Fig. 5. Example of object detection using TM Vision.

IV. Experiment

In this assembly experiment, the robots cooperate to assemble a planetary gear. The parts are picked up and delivered to the robots by an autonomous mobile robot. In addition, Robot 2 is equipped with 3D printed customized gripper extensions in order to get a good grip on circular objects. The mobile robot is first sent to the inventory station to pick up the parts used in this assembly. In the mean time, the TM robots localize each other as described in section III-C. Robot 2 receives a message from the mobile robot after it has arrived at the assembly area. Robot 2 then scans the area where the mobile robot is standing for a landmark. When the landmark is found the Robots will move to the left of the landmark and scan the QR code located on the part tray. The QR tray describes which tray this is. If the robot scans the wrong tray, the robot will move to the other tray and repeat the process. This ensures that the robot is at the correct tray before proceeding with the program, as described in Algorithm 1.

Once the correct tray is scanned, Robot 2 proceeds to pick up the planet carrier. It then sends a message to let the mobile robot move to Robot 1. Robot 1 scans for the landmark to orient itself. It also scans the QR code to verify that the tray is correct. When the correct tray is confirmed Robot 1 moves above the fixed position where the planet gears are placed on the tray and uses image recognition to pick them up, as seen in Fig. 5. Fig. 6 depicts the steps of inserting the planet



(a) Pick up planet carrier
(b) Locate planet gears
(c) Hover pin
(d) Insert planet gear and release
(e) Repeat planet gear insertion
(f) Insert sun gear

Fig. 6. Screenshots of steps for inserting planet gears.

gears. Robot 1 first moves to a hovering position above the planet gear, picks up the first planet gear, and moves to the collaborative work area. Prior to inserting the planet gears on the pins, Robot 1 takes a picture above the planet gear carrier and uses image processing to find the location of the pins. The robot then moves down, hovering the first pin to place the first planet gear. Robot 1 then proceeds with the "Smart Insert" function depicted in Algorithm 2. When the force limit is reached, the robot will push the planet gear downwards until the force limit or distance limit is reached. The same process is repeated for the rest of planet gears.

The next is to insert the sun gear. In order to fit the sun gear in between the three planet gears, the teeth on the planet gears must be aligned with the teeth on the sun gear. This is done by tilting the sun gear and inserting it from above. Fig. 7 shows an example plot of force and torque from the force sensor. When the sun gear hit the planet gears, the robot continues by dragging the sun gear backwards in between two planet gears. This will fit the sun gear with two of the three gears. The last step is to tilt the sun gear back up while rotating and keeping the tip of the sun gear in the middle of the planet gears. This will align the last planet gear and the robot can release the sun gear.

The function that is used to lock the lid in place has two parameters: torque limit and distance limit. The limits are set to decide when the function is considered complete, i.e., either it meets a certain counterforce, or when the rotational movement reaches the limit. When attaching the lid on the sun gear to the pins on the carrier, a couple errors may occur. The error that occurs most frequently is that the lid is released prior to the pins and the pinholes has been aligned. The robot will then release the lid without the lid falling down with the pins in the pinholes. This causes the gear to not be properly attached and fall on the ground and break. This was solved using the different outputs from the "Grip" function. By utilizing this combined with the torque limit, the robot manages to lock the lid in place consistently.

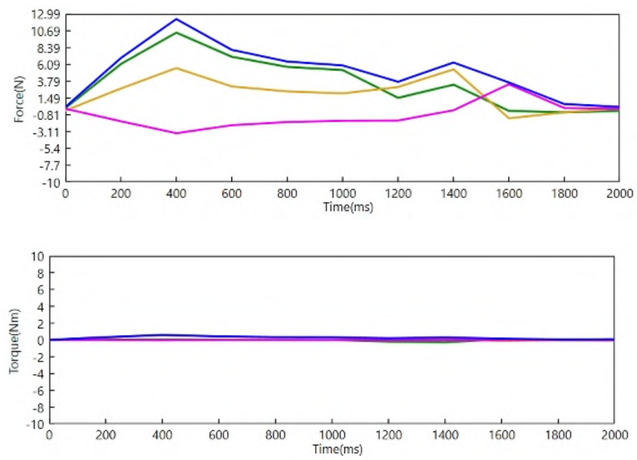
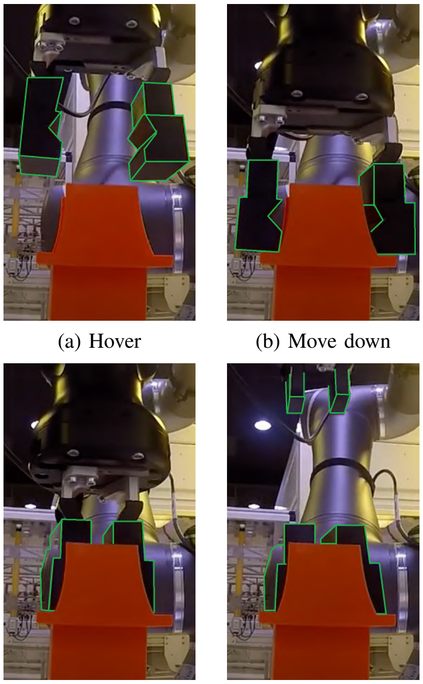


Fig. 7. Example of force/torque over time in TM Flow.

Once the lid has been locked successfully, Robot 1 grips the gear assembly and sends a message to Robot 2, indicating that the assembly can proceed. Robot 2 releases the gear assembly and turns around to remove the gripper extensions, as shown in Fig. 8. This is done by hovering a gripper extension remover, moving down and gripping. This results in the gripper extensions keeping inside the housing on the gripper extension remover. Once the gripper extensions are locked in the housing, the robot moves straight up, leaving the extensions in the housing.

At the last step, Robot 2 picks up the ring gear from the mobile robot, moves back to the collaborative work area, and sends a message to Robot 1 indicating that the assembly can continue. Robot 1 then starts by using the ring gear to position the internal gear module by pushing it softly on the outer edge of the ring gear. After the internal gear module has been positioned correctly, Robot 1 begins to insert the gear module



(a) Hover (b) Move down

(c) Grip (d) Detach

Fig. 8. Steps for detaching gripper extensions.

into the ring gear. This is done in a similar fashion to the process explained in Algorithm 2.

When the gear is fully assembled, Robot 2 moves back to the mobile robot and places the gear into a socket on the part tray. A message is sent to the mobile robot when this process is complete. The mobile robot then drives away with the part and the assembly is complete. A full video can be seen at: https://www.youtube.com/watch?v=_xkODty-A94.

V. Conclusion and future works

In this study, an autonomous system for complex assembly has been presented. An assembly of planetary gears has been taken as an example to verify the effectiveness of the system. Using 3D printed gears as test-objects, the concept was tested using two collaborative robots to assemble the gear, and a mobile robot to deliver the different parts. In this scenario, tight tolerances demands high precision and repeatability, which makes sensor input from vision camera and force sensors important.

The experimental results show that using robots in complex assembly tasks can be beneficial and relieve human workers of repetitive and strenuous tasks. By utilizing more precise sensors and cameras such as 3D-cameras, it could give more precise data which could improve the manufacturing process further. Furthermore, by including a fleet of mobile robots, a completely autonomous process can be achieved.

Acknowledgment

Thanks to Ola Jon Mork and Irina-Emily Hansen at Department of Ocean Operations and Civil Engineering at NTNU for insightful comments. Also, thanks to Jonas Borgundvåg at Omron for technical support.

References

[1] L. S. Dalenogare, G. B. Benitez, N. F. Ayala, and A. G. Frank, "The expected contribution of industry 4.0 technologies for industrial performance," *International Journal of Production Economics*, vol. 204, pp. 383–394, 2018.

[2] R. Y. Zhong, X. Xu, E. Klotz, and S. T. Newman, "Intelligent manufacturing in the context of industry 4.0: a review," *Engineering*, vol. 3, no. 5, pp. 616–630, 2017.

[3] H. Boyes, B. Hallaq, J. Cunningham, and T. Watson, "The industrial internet of things (iiot): An analysis framework," *Computers in industry*, vol. 101, pp. 1–12, 2018.

[4] L. I. Hatledal, R. Skulstad, G. Li, A. Styve, and H. Zhang, "Co-simulation as a fundamental technology for twin ships," *Modeling, Identification and Control*, vol. 41, no. 4, pp. 297–311, 2020.

[5] J. Wan, S. Tang, Z. Shu, D. Li, S. Wang, M. Imran, and A. V. Vasilakos, "Software-defined industrial internet of things in the context of industry 4.0," *IEEE Sensors Journal*, vol. 16, no. 20, pp. 7373–7380, 2016.

[6] G. Li, H. P. Hildre, and H. Zhang, "Toward time-optimal trajectory planning for autonomous ship maneuvering in close-range encounters," *IEEE Journal of Oceanic Engineering*, vol. 45, no. 4, pp. 1219–1234, 2019.

[7] G. Li, H. Zhang, J. Zhang, and H. P. Hildre, "An approach for adaptive limbless locomotion using a cpg-based reflex mechanism," *Journal of Bionic Engineering*, vol. 11, no. 3, pp. 389–399, 2014.

[8] R. Ma, J. Chen, and J. Oyekan, "A review of manufacturing systems for introducing collaborative robots," in *3rd UK-RAS Conference for PhD Students & Early Career Researchers*, 2020, pp. 71–73.

[9] Q. Ju and F. Pan, "Performance comprehensive assessment system for the manufacturing system of automotive assembly line," in *2016 Chinese Control and Decision Conference (CCDC)*, 2016, pp. 4672–4676.

[10] W. Wan, "Using intelligent robots to assemble automobile parts," *Advances in Automobile Engineering*, vol. 6, no. 01, pp. 1–2, 2017.

[11] S. Smys and G. Ranganathan, "Robot assisted sensing, control and manufacture in automobile industry," *Journal of ISMAC*, vol. 1, no. 3, pp. 180–187, 2019.

[12] Omron. Collaborative robot for assembly, packaging, inspection and logistics. [Online]. Available: https://d3mm496e6885mw.cloudfront.net/manufacturer_product/5c37be7e3d75cb67d13da4a4/specsheet/specSheets/original/CollaborativeTM1214DatasheetEN201810R274IE01.pdf

[13] ——. Software manual tmflow. [Online]. Available: https://assets.omron.eu/downloads/manual/en/v1/tm_flow_software_manual_installation_manual_en.pdf

[14] ——. Software manual tm vision. [Online]. Available: https://assets.omron.eu/downloads/manual/en/v5/tm_vision_software_manual_installation_manual_en.pdf

[15] Robotiq. Robotiq ft 300 force torque sensor for omron tm series robots. [Online]. Available: https://assets.omron.eu/downloads/manual/en/v2/robotiq_ft_300_force_torque_sensor_for_omron_tm_series_robots_instruction_manual_en.pdf

[16] Omron. Software manual mobile robot. [Online]. Available: https://assets.omron.com/m/0c55a28223f13734/original/Mobile-Robot-Software-Suite-User-s-Guide.pdf

[17] S. Thrun, "Simultaneous localization and mapping," *Robotics and Cognitive Approaches to Spatial Mapping*, 2007.

[18] Omron, *Advanced Robotics Command Language, Mobile Robots*, Omron, 2017.