An Operational 5G Edge Cloud-Controlled Robotic Cell Environment based on MQTT and OPC UA

Arias-Cachero Rincón, David; Celik, Ali Ekber ; Zhang, Weifan; Rodriguez, Ignacio; Yavuz, Sirma Cekirdek; E. Mogensen, Preben

Published in:
2023 21st International Conference on Advanced Robotics (ICAR)

Publication date:
2023

Document Version
Accepted author manuscript, peer reviewed version

An Operational 5G Edge Cloud-Controlled Robotic Cell Environment based on MQTT and OPC UA

David Arias-Cachero Rincon[1,2], Ali Ekber Celik[3,4], Weifan Zhang[2],
Ignacio Rodriguez[1], Sirma Çekirdek Yavuz[4], and Preben Mogensen[2]

***Abstract*— This paper presents the design and implementation of a wireless robotic cell environment based on 5G and edge-cloud technologies. In the process, different control communication protocols and architectures concerning the MQTT and OPC UA IIoT standards are considered based on available reference implementations. The industrial-grade robotic cell is evaluated in operational conditions with a focus on the specifics of the underlying control communication effects, as well as on the impact on the industrial manufacturing use case. The experimental results demonstrated the effectiveness of the wireless 5G edge-cloud operation using both MQTT and OPC UA protocols, providing the required flexibility and re-configurability, at expenses of an 8-9% increased production cycle time, as compared to the wired Ethernet reference.**

I. INTRODUCTION

The fifth-generation wireless cellular technology (5G) is settled to play a big role in the development of Industry 4.0 [1]. Combined with Edge-Cloud (EC) computing, 5G can be applied to different Cyber-Physical Systems (CPS), leading to versatile Industrial Internet-of-Things (IIoT) solutions, which will be the core of the highly flexible and re-configurable facilities in the factories of the future [2]. By applying 5G Edge-Cloud (5G-EC), it will be possible to replace cables and enable new industrial use cases requiring mobility support [3]. Moreover, 5G-EC will also facilitate "cloudification", allowing the migration of computing resources currently used in factory shop floor devices to the cloud infrastructure. This will trigger the introduction of optimized manufacturing control architectures based on EC-centralized management [4].

Assuming the simplest design of an industrial robotic cell [5], including one robotic manipulator arm controlled from a wired Programmable Logic Controller (PLC), further integrated into the cabled network of a factory to connect the cell with the high-level production management tools, e.g., Manufacturing Execution System (MES); 5G-EC can be used to remove the PLC and migrate its intelligence to the cloud, establishing a direct wireless communication between the robotic manipulator and the management systems. This architectural revolution is illustrated in Figure 1, and it comes with a number of associated benefits:

- The cables between the PLC and the robotic arm, and between the PLC and the high-level management are removed: this enables flexibility in the physical configuration of the robotic cell layout and triggers operational optimization possibilities [5].
- Having a "cloudified", virtualized, and containerized software version of the PLC allows for centralized control re-configuration, eliminating the need of programming and debugging with physical access to the line PLCs. Further, this facilitates the possibility of rapidly manufacturing custom solutions [3].
- The centralized 5G-EC architecture facilitates the information exchange as it enables seamless information sharing and coordination among various manufacturing elements [6].

The control and operation of a robotic cell is not only a matter of communication network architecture; control protocols are also important [1]. In recent years, there has been a growth in use of specific protocols targeting the IIoT space, by addressing Information Technology (IT)/Operational Technologies (OT) convergence, aiming at facilitating the monitoring and control of industrial applications [7]. Message Queuing Telemetry Transport (MQTT) and Open Platform Communications Unified Architecture (OPC UA) are two of the most important and widespread ones [8]. In this respect, while there is some literature analyzing, comparing, and addressing the performance of these protocols in standard wired network settings [8], [9], there is still a gap in the analysis and evaluation over 5G-based configurations [7]. Further, works in the literature typically focus only on the communication perspective and miss the evaluation of the impact of these on the actual industrial manufacturing-relevant aspects, e.g., production cycles and production throughput.

To address those gaps, in this paper, we focus on the design, and operational implementation of a wireless industrial network solution to enable the control of a robotic cell

*This work was partially supported by Innovation Fund Denmark through the project "5G-ROBOT - 5G Enabled Autonomous Mobile Robotic Systems"; and by the Spanish Ministry of Science and Innovation under Ramon y Cajal Fellowship number RYC-2020-030676-I funded by MCIN/AEI/10.13039/501100011033 and by the European Social Fund "Investing in your future".

[1]David Arias-Cachero Rincon, and Ignacio Rodriguez are with Area of Signal Theory and Communications, Department of Electrical Engineering, University of Oviedo, Gijon, 33203, Spain {uo264705, irl}@uniovi.es

[2]David Arias-Cachero Rincon, Weifan Zhang, and Preben Mogensen are with the Wireless Communication Networks Section, Department of Electronic Systems, Aalborg University, Aalborg Øst, 9220, Denmark {dacr, weifanzh, pm}@es.aau.dk

[3]Ali Ekber Celik is with Universal Robots A/S, Odense SØ. 5220, Denmark aece@universal-robots.com

[4]Ali Ekber Celik and Sirma Çekirdek Yavuz are with the Department of Computer Engineering, Yildiz Technical University FBE, Istanbul, 34220, Turkey ekber.celik@std.yildiz.edu.tr, smyavuz@yildiz.edu.tr

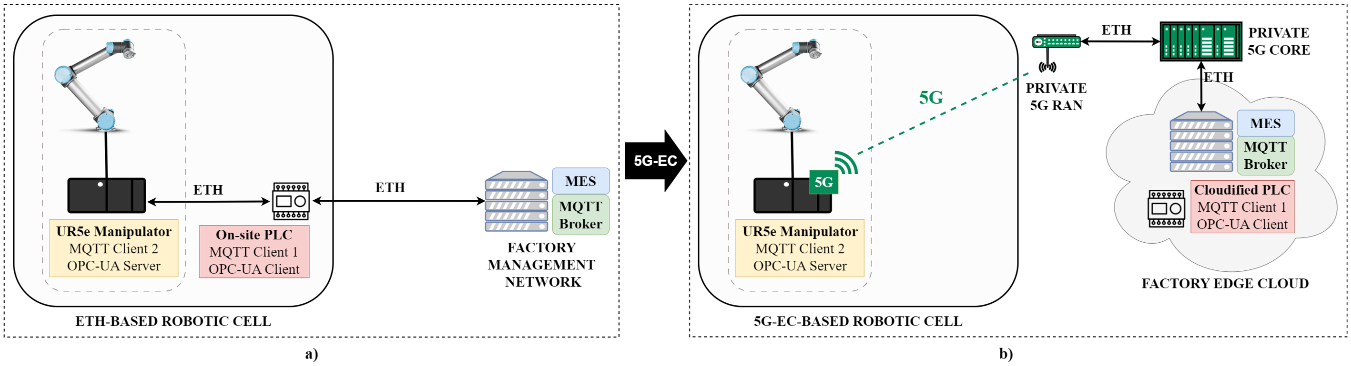

Fig. 1. High-level architecture of the implemented robotic cell environment: a) over ETH (reference case), b) over 5G-EC.

environment from a remote PLC using 5G-EC technology. As part of the implementation, control communication over MQTT and OPC UA is considered in order to shed some light of the performance of the protocols based on the same reference implementation of the selected industrial use case. A thorough performance evaluation was carried out in realistic industrial settings using commercial-grade robotic equipment over an enterprise-grade Private 5G-EC deployment. As part of the evaluation, the exact same implementation was operated over Ethernet (ETH) cabled settings, to obtain a performance baseline reference, which would allow to quantify the effect of wirelessly operating the industrial use case over 5G-EC. The performance evaluation of the industrial use case considers both communication-related aspects and production-related aspects, allowing for detailed insights and observations at all operational levels.

The rest of the paper is organized as follows: Section II introduces the basics of the main IIoT communication protocols chosen for the implementation. Section III describes the operational robotic cell environment and configuration, as well as its integration with the 5G-EC, and the methodology followed for performance testing and evaluation. Section IV presents the performance results of the robotic cell when controlled based on MQTT and OPC UA protocols and operated over 5G-EC. Further, the results are discussed and compared with baseline ones obtained by the same robotic cell environment operated over wired ETH. Finally, Section V concludes the paper.

II. IIoT Communication Protocols

Prior to focusing on implementation and evaluation aspects, the main characteristics of the MQTT and OPC UA IIoT protocols are surveyed, and the choice of specific protocol configurations is motivated.

A. MQTT

MQTT is a lightweight IIoT broker-based protocol [10]. The exchange of data happens between devices following a Publisher-Subscriber (PubSub) paradigm. The MQTT clients that publish data are called publishers whereas the ones interested on receiving that data are the subscribers. Messages are filtered by the broker within topics, and if a client is interested in a specific flow, it needs to subscribe to it. One of the most important features of this protocol is that is it offers different Quality of Service (QoS) schemes [10]. Even though MQTT is Transmission Control Protocol (TCP)-based at transport layer, which already guarantees delivery, the MQTT QoS schemes build an extra level of reliability and redundancy at application level. There are three different configurations of QoS:

- QoS 0: also called "at most once", is a best-effort delivery mechanism. There are not application layer Acknowledgments (ACKs) or any other control techniques that guarantee application layer reliability.
- QoS 1: also called "at least once". It adds ACKs for the application layer transmissions. Here, the exchange of packets may lead to the reception of duplicates. This configuration guarantees that the subscriber receives the application data at least one time.
- QoS 2: also called "exactly once". It guarantees reliability of the communication avoiding scenarios with duplicated messages. To achieve this, there is a substantial exchange of packets between the MQTT clients.

There are also User Datagram Protocol (UDP)-based versions of MQTT, such as MQTT for Sensor Networks (SN) [11]. It enables efficient communication to devices with limited memory, processing power and network capabilities; at expenses of the inability of detecting if any data has been lost at transport and application layers. Therefore, MQTT should be only used in use cases that can tolerate a certain amount of packet loss, i.e., non-critical video transmissions.

B. OPC UA

OPC UA is famous industrial communication standard used to exchange data between different devices within industrial automation scenarios. It was created as a measure to integrate industrial equipment from different manufacturers and vendors at all levels of the automation pyramid [12], boosting the data collection, scalability, and security of industrial control ecosystems. OPC UA is implemented with devices that can take the role of OPC UA clients or OPC UA servers and exchange information data following two different communication paradigms: Client-Server and PubSub [13]. In the first one, an OPC UA Client requests a resource from an OPC UA Server; whereas in the second one, an OPC UA client subscribes to a resource stored on the OPC UA server and specifies the scheduling period/rate at which it wants to be notified if there is a variation in the resource that it has subscribed to.

The most common implementations of OPC UA are over TCP. However, OPC UA can be combined with other underlying communication schemes and can be operated over other protocols such as UDP, and even MQTT [13]. This is because OPC UA is more than an application layer protocol. OPC UA is an architecture, where data is organized in standardized namespaces that are information models composed by variables (basic data elements), methods (actions applied to a variable), and events (alerts triggered when a condition is met).

C. Choice of IIoT Communication Protocols for Implementation

For our assessment, two main configurations have been chosen for implementation and evaluation: MQTT QoS 0 and OPC UA PubSub with a PubSub scheduling period of 10 ms (PubSub:10ms). The reason for this choice is that MQTT QoS0 will avoid all unnecessary overhead and redundant exchange of packets introduced by the QoS 1 and QoS 2 configurations. In the case of OPC UA, the PubSub scheme is selected for two reasons: first, it is a more scalable version of the protocol as compared to the server-client one; second, it makes it similar and more comparable to the MQTT QoS0 as both are PubSub-based. For OPC UA, the PubSub refreshing period needs to be set to values of 1 ms and above. In our case, a 10 ms rate is chosen as it provides a good trade-off between the refresh rate of the protocol and the 5G network scheduling rate configuration. Moreover, the physical movements of the robotic arm in the implemented application are not as fast as 1 ms, and therefore a more relaxed period is suitable in this specific industrial use case. The choice of OPC UA PubSub rate was also validated by the vendor of the robot as it matches closely the configured sensor-actuator rate of the operational robotic arm.

III. OPERATIONAL ROBOTIC CELL ENVIRONMENT BASED ON 5G EDGE-CLOUD

A. Physical Setup and Control Communication Architecture

The 5G-EC implementation of the robotic cell for this study was carried out at the AAU 5G Smart Production Lab, a operational small factory research lab [14], as per the high-level architecture depicted in Figure 1.b, based on the following main elements:

- UR5e robotic manipulator arm [15]: light payload industrial robotic arm with 6 rotating joints/Degrees of Freedom (DoF).
- 5G Box [16]: industrial-grade connectivity gateway with ETH-5G capabilities based on SIM8262E-M2 modems [17].
- Private 5G network: fully private industrial 5G network operating in Stand-Alone (SA) Rel.15 mode, composed of an on-site private 5G core and 3 Radio Access Network (RAN) base stations configured with 100 MHz of bandwidth, and 3/7 uplink/downlink Time Division Duplex (TDD) ratio [18].
- Edge cloud platform [19]: enterprise cloud comprised of 12 servers, directly connected to the private 5G core, each of them with 56 CPU Logical cores, 125 GiB of RAM, and 932 GiB of storage.
- Cloudified PLC: software instance of a physical PLC which instructs the control of the UR5e robotic arm manufacturing process. It is executed in a virtual machine with access to 32 CPU logical cores, 60 GiB of RAM, and 128 GiB of storage.

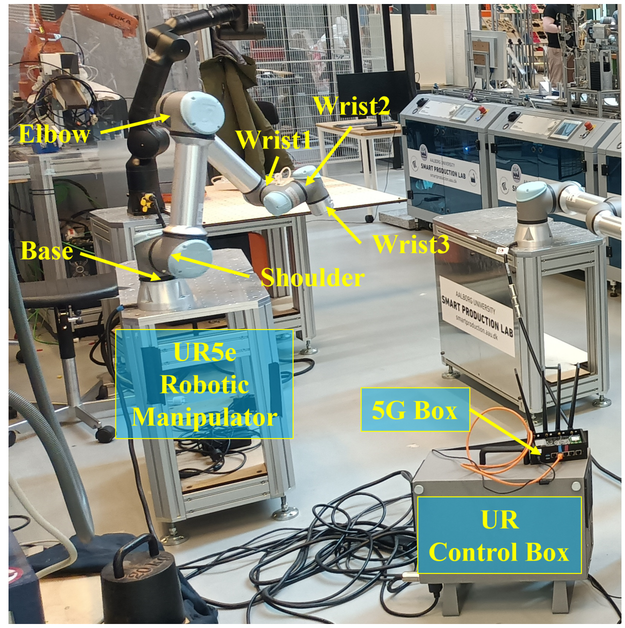

Fig. 2. Picture of the operational robotic cell environment implemented at the AAU 5G Smart Production Lab based on a 5G-integrated UR5e robotic manipulator arm.

Figure 2 shows a picture of the implemented robotic cell environment. In terms of integration of the robotic cell elements with the Private 5G network, a "bridge" is created between the robot control box and the 5G Box, which guarantees 5G connectivity and access to the EC network at which the PLC software is deployed as a virtualized entity. Please, note that it is the PLC what is virtualized, not the robot controller. The specific software networking and routing details are given in [20]. The baseline reference wired ETH architecture displayed in Figure 1.a was also implemented for evaluation and comparison in terms of performance with the 5G-EC one. In this case, the robot control box was directly cabled to the EC platform, allowing for a 1-to-1 comparison of both solutions, as the only different element is the underlying communication technology (ETH or 5G) and all hardware and software computing elements remain the same. It should be noted that the routing of control information is exactly the same in both cases, as the 5G-EC "bridge" that replaces the ETH connection is totally transparent to the robotic control elements.

The control communication of the robotic cell between the PLC and the UR5e robotic manipulator was implemented

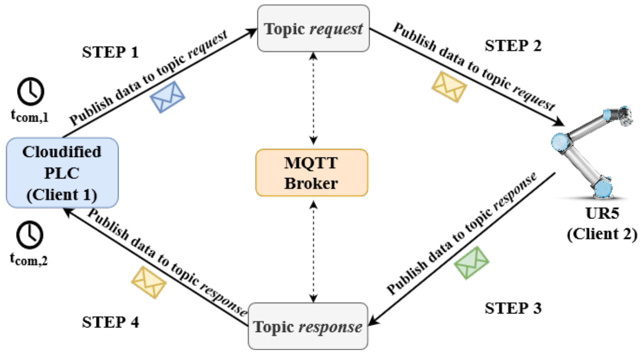

Fig. 3. Diagram of steps and entities involved in the exchange of control messages with MQTT QoS0.

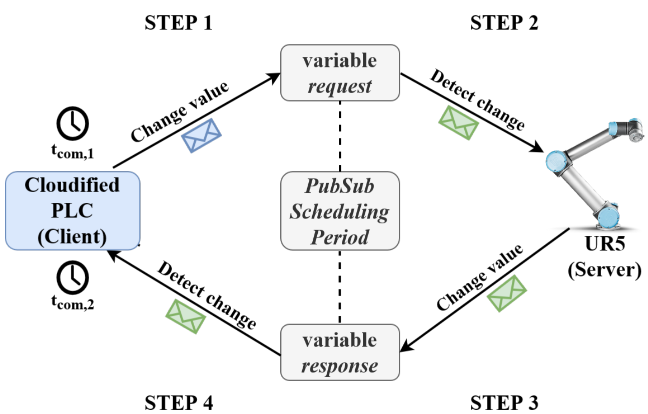

Fig. 4. Diagram of steps and entities involved in the exchange of control messages with OPC UA PubSub.

based on the exchange of control messages as described in Figures 3 and 4 for the selected MQTT QoS0 and OPC UA PubSub:10ms protocols, respectively. The implemented closed control loop has the purpose of sending control commands from the PLC to the UR5e robotic manipulator and obtain a status reply back from the robotic arm.

In the case of MQTT QoS0, the exchange of control messages involves three entities: the cloudified PLC (MQTT Client 1) and the MQTT broker, both deployed in software in the EC; and the UR5e manipulator arm (MQTT Client 2). Here, Client 1 publishes control data to the topic "request" to the broker and subscribes to the topic "response" in order to get the status message back. Client 2 does the opposite, subscribes to the topic "request" and publishes data to the topic "response". All information exchange between clients is managed by the broker.

When it comes to the OPC UA PubSub:10ms scenario, the closed control loop is slightly different. In this case, since there is no broker, to send a control command, the cloudified PLC (OPC UA client) modifies the content of the variable "request" to which the UR5e robotic arm (OPC UA server) is subscribed to. When the server detects a change in the value of "request", it acts over the variable "response" (to which the client is subscribed to) with status information. Subscription are updated at the configured PubSub scheduling rate of 10 ms.

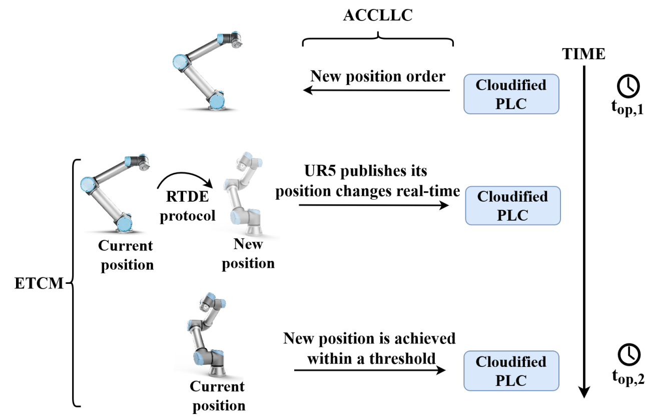

Fig. 5. Diagram of the operational behavior of the robotic manipulator from control command issuing to movement execution, completion, and notification.

Fig. 6. Diagram of the operational manufacturing cycle configured including computing and movement execution slots.

B. Use Case Configuration

As an implementation demo use case, the robotic manipulator was configured to perform a simple hammering action. To do that, The PLC sends command orders in which the payload data contains an array of 6 floating values in the format [base, shoulder, elbow, wrist1, wrist2, wrist3] representing the target position in radians for each of the arm joints. Once the command is received by the robotic manipulator using MQTT or OPC UA, the packet payload is decoded and acquired using the Real-Time Data Exchange (RTDE). Next, the data is applied to the joints, so the robot starts to move towards the target position. During this process, the PLC is subscribed to the robotic arm status messages, so it receives the current position of the arm in real-time. Each of the data received by the PLC is compared with an accuracy threshold of 8.75×10^{-4} radians and, once the threshold is achieved, the movement is considered as completed, so the PLC sends a new command, repeating the previous steps. This process is illustrated in Figure 5. The control commands and associated status replies have a data size of 120-134 B, which is written as an application layer payload to the IIoT communication protocols.

One operational manufacturing cycle of the hammering action implemented is achieved by the issuing of 10 control commands and execution of the associated movements, as detailed in Table I. These are executed sequentially as illustrated in Figure 6, leading to cycle times and overall production throughput that are dependent on the computing, communication, and physical movement of the robotic arm.

TABLE I

LIST OF CONFIGURED COMMANDS AND MOVEMENTS OF THE ROBOTIC MANIPULATOR IN ONE OPERATIONAL MANUFACTURING CYCLE

Action	Description	Value
Control command 1	[0.125664, -2.07083, -2.20086, -0.39235, 1.459793, 0.890642]	
Movement 1	Wake-up (only first time, then idle)	go to rest position
Control command 2	[0.125664, -1.80903, -2.20086, -0.39235, 1.459793, 0.890642]	
Movement 2	Shoulder elevation	shoulder lifted from rest position
Control command 3	[-1.51058, -1.80903, -2.20086, -0.39235, 1.459793, 0.890642]	
Movement 3	Base rotation forward	base rotated clockwise (94 degrees rotation)
Control command 4	[-1.51058, -1.80903, -1.102, -0.39235, 1.459793, 0.890642]	
Movement 4	Elbow elevation	elbow lifted to prepare for hammering action (63 degrees rotation)
Control command 5	[-1.51058, -1.80903, -1.102, -0.23754, 1.459793, 0.890642]	
Movement 5	Hammer down	wrist 1 action (9 degrees rotation)
Control command 6	[-1.51058, -1.80903, -1.102, -0.23754, 1.459793, 0.890642]	
Movement 6	Idle	-
Control command 7	[-1.51058, -1.80903, -1.102, -0.39235, 1.459793, 0.890642]	
Movement 7	Hammer up	wrist 1 retraction
Control command 8	[-1.51058, -1.80903, -2.20086, -0.39235, 1.459793, 0.890642]	
Movement 8	Elbow drop	elbow lowered to to rest position
Control command 9	[0.125664, -1.80903, -2.20086, -0.39235, 1.459793, 0.890642]	
Movement 9	base rotation backward	base rotated counter-clockwise 94 degrees
Control command 10	[0.125664, -2.07083, -2.20086, -0.39235, 1.459793, 0.890642]	
Movement 10	Shoulder drop	shoulder lowered to rest position
→ new cycle → Control command 1		

C. Key Performance Indicators and Testing Procedures

The testing and evaluation of the performance of the deployment considered aspects in both the communication and the industrial manufacturing domains. The assessment was based on the following Key Performance Indicators (KPIs):

- Closed Control Loop Latency (CCLL): characterizes the impact of the IIoT communication protocols on the closed control loop considering the behaviour of the underlying communication layers (from physical layer to application layer). This is done by evaluating the time that it takes between the issuing of a specific control packet at the PLC side until a response to that packet is released by the robotic arm and received back at the PLC, i.e., the total time elapsed in the execution of all 4 steps indicated in Figures 3 and 4, calculated as per Equation 1.

$$CCLL \text{ [ms]} = t_{com,2} \text{ [ms]} - t_{com,1} \text{ [ms]} \tag{1}$$

- Operational Closed Control Loop Latency (OPCCLL): evaluates the impact of the underlying IIoT communication protocols at operational manufacturing level by characterizing the time elapsed since a specific control command is issued by the PLC until the action is physically performed by the robotic manipulator and acknowledged back to the PLC. The main difference between OPCCLL and CCLL is that OPCCLL considers the Accumulated CCLL of all the Communications (ACCLLC) exchanged between the PLC and the robotic entity in the high-level control process, as well as the physical Elapsed Time for the Current Movement (ETCM). Mathematically, is computed as indicated in Equation 2, as per the reference of the different parameters in Figure 5.

$$\begin{aligned} OPCCLL \text{ [ms]} &= ACCLLC \text{ [ms]} + ETCM \text{ [ms]} \\ &= t_{op,2} \text{ [ms]} - t_{op,1} \text{ [ms]} \end{aligned} \tag{2}$$

- Operational Cycle Elapsed Time (OPCET): quantifies the impact of the underlying IIoT communication protocols on the operational manufacturing cycle. Differently from OPCCLL which focused on single robotic actions, OPCET considers the combination of control communications, physical movements of the robotic arm, and transitional computing processing times (t_c) in a complete production cycle, i.e., in the 10 sequential programmed actions including the 5 forward ones to perform the hammering, as well as the 5 backwards ones to return to the initial home position. This is described by Equation 3 and the reference parameters in Figure 6.

$$\begin{aligned} OPCET \text{ [ms]} &= \sum_{i=1}^{10} [OPCCLL(i) \text{ [ms]} + t_c(i) \text{ [ms]}] \\ &= t_{cycle,2} \text{ [ms]} - t_{cycle,1} \text{ [ms]} \end{aligned} \tag{3}$$

In order to evaluate the proposed KPIs, different tests were executed. CCLL was evaluated by considering 500 consecutive communication closed control loops for the different MQTT QoS0 and OPC UA PubSub:10ms protocols considering separately ETH and 5G-EC as underlying technologies. This evaluation was done considering two different payload size configurations (2 B and 1300 B) as well as multiple network conditions with variable load (ranging from 0% to 100%). The chosen payload sizes allow to characterize the operational performance range of the IIoT protocols, as the performance for 2 B payloads is representative of the one from control systems where few data is needed, while the 1300 B payload performance is representative of the one from the most demanding control cases where a lot of information is exchanged. This is due to the fact

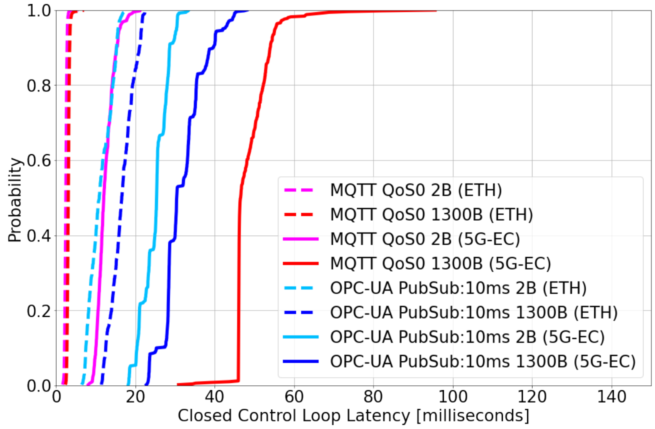

Fig. 7. CCLL performance for the different IIoT protocols and communication technologies with variable payload and 100% network load.

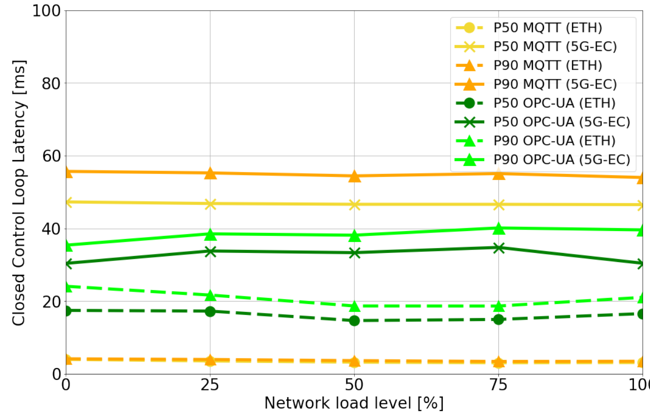

Fig. 8. CCLL performance for the different IIoT protocols and communication technologies with 1300 B of payload and variable network conditions.

that a payload data size of 1300 B in application layer, corresponds to a frame size of close to 1500 B at data-link layer, which translates into a full Maximum Transfer Unit (MTU) size. This is a worst-case scenario assuming a standard ETH network, and also in 5G-EC as, typically, over wireless, the smaller the packages the better the performance. It is expected that implementations operated with intermediate payload sizes will result in CCLL performance results contained within the performance boundaries set by the 2 B and 1300 B results. The ETH network was loaded by adding iPerf3-based UDP background traffic [21], generated from an external device, proportionally to a maximum network capacity of 1000 Mbps. In the 5G-EC case, the Private 5G wireless network offered asymmetric bandwidth [18] with 550 and 98.5 Mbps in Downlink (DL) and Uplink (UL), respectively. Therefore, the load of the 5G-EC network generated by an external 5G device was set individually in each link direction (in DL from EC to robotic arm, in UL from robotic arm to EC) proportionally to the maximum capacities.

OPCCLL and OPCET were evaluated by gathering the results from 40 consecutive full operational manufacturing cycles, resulting, i.e., in 40 OPCET samples and 400 OPCCLL samples. As in the CCLL case, this evaluation considered the different MQTT QoS0 and OPC UA PubSub:10 ms protocols separately operated over ETH and 5G-EC. In this case, the payload is not a variable. As these tests consider operational manufacturing conditions, the payload was determined by the specific use case requirements and control data defined in Section III.B, e.g., 120-134 B.

IV. PERFORMANCE RESULTS

Figure 7 depicts the CCLL performance results of MQTT QoS0 and OPC UA PubSub:10ms for payload sizes of 2 B and 1300 B and network load of 100% in the shape of Empirical Cumulative Density Functions (ECDFs). The figure includes results for both the baseline wired ETH case and the 5G-EC case, providing a bounded reference of the performance range expected with the analyzed IIoT protocols in loaded network conditions. The results indicate that for the

TABLE II
SUMMARY OF CCLL PERFORMANCE RANGE IN MS FOR THE DIFFERENT IIOT PROTOCOLS AND COMMUNICATION TECHNOLOGIES

Technology	MQTT QoS0	OPC UA PubSub:10ms
ETH	2.5-10.9	10.1-16.6
5G-EC	12.1-46.5	25.3-39.6

selected IIoT protocols, the MQTT and OPC UA median CCLL performance over ETH with the small 2 B payload is 2.5 and 10.1 ms, respectively. This slightly increased to 10.9 ms for MQTT and 16.6 ms for OPC UA in the case of the large 1300 B payload. These values are well aligned with those previously reported in the literature [8], [9]. For the 5G-EC cases, MQTT leads to a median CCLL performance of 12.1 ms for the 2 B payload and 46.5 ms for the 1300 B one. OPC UA performance is slightly increased as compared to MQTT for 2 B with 25.3 ms. However, for the 1300 B case, the OPC UA median performance is 39.6 ms, which is approximately 7 ms better than the MQTT one over 5G-EC. These results highlight the impact of large packet sizes on 5G-EC performance - as anticipated in Section III.C, and second, the importance of verifying the IIoT protocols in all potential operational conditions prior to deployment. For quick reference, CCLL performance ranges are summarized in Table II.

In Figure 8, median (P50) and 90th-percentile (P90) CCLL values are displayed for the most demanding cases with 1300 B payload for variable network load conditions. As detailed, both the MQTT and OPC UA protocols scale well with network conditions, as the CCLL performance is similar and stable for all studied network loads. From the full CCLL analysis presented, it can be concluded that the payload configuration is the technical aspect which might impact the most the performance of the IIoT communication protocols, especially when operating MQTT over 5G-EC.

From an industrial manufacturing perspective, Figure 9 illustrates the OPCCLL performance results for the configured robotic manipulator arm movements when operated over the MQTT QoS0 and OPC UA PubSub:10ms protocols, for both

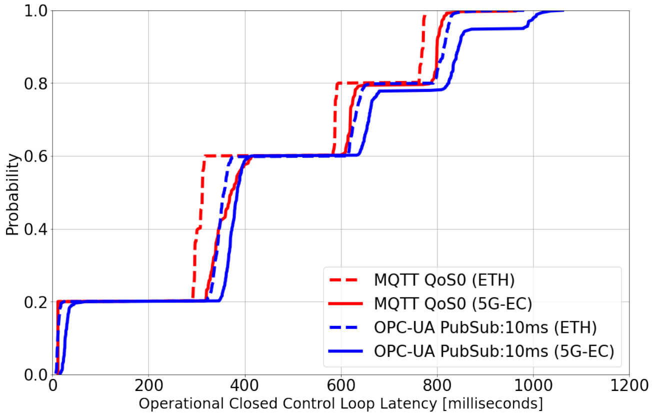

Fig. 9. OPCCLL performance for the different IIoT protocols and communication technologies in industrial operational conditions.

TABLE III
SUMMARY OF MEDIAN OPCCLL VALUES IN MS FOR THE DIFFERENT IIOT PROTOCOLS AND COMMUNICATION TECHNOLOGIES

Movement pair	1-6	2-10	3-9	4-8	5-7
MQTT QoS0 (ETH)	12	312	771	588	296
MQTT QoS0 (5G-EC)	12	372	800	620	333
Δ_{MQTT}	0	60	29	42	37
OPC UA PubSub:10ms (ETH)	11	338	815	630	357
OPC UA PubSub:10ms (5G-EC)	27	365	835	650	385
Δ_{OPC-UA}	16	27	20	20	28

ETH and 5G-EC. As observed, each of the ECDFs present 5 distinguishable steps, one per pair of the programmed movements of the robot, i.e., movements 2, 3, 4, and 5 are paired to movements 10, 9, 8, and 7, respectively, as they performed a similar (but opposite) movement. Idle movement 1 and 6 are also paired in this respect, as no movement is done. It should be noted that two of the pairs of movements (2-10 and 5-7) are very similar in performance, with an OPCCLL of approximately 300-400 ms and, thus, they partially overlap in the statistical representation. Median values for each of the pairs of movements, protocols, and underlying communication technology are summarized in Table III. The difference in performance for the different movements is due to the fact that some of them require large physical transitions, e.g., movements 3 and 9 (base rotation), resulting in an OPCCLL of approximately 800 ms; while others imply no movement, e.g., idle actions, where only control command communication is exchanged, resulting in an OPCCLL of approximately 15 ms. On average, the OPCCLL performance is 30.4 ms better for MQTT than for OPC UA. For both IIoT protocols, the steepness of the curves, indicates that ETH produces slightly more stable (repetitive) performance values than 5G-EC. In the OPC UA case, there are some outliers in the upper part of the 5G-EC curves, which indicate that, during the operational evaluation over 5G-EC, some of the movements exhibited such increased latency during approximately 5% of the test.

The table includes the parameters Δ_{MQTT} and Δ_{OPC-UA}, both in ms, which quantify the gap in

TABLE IV
SUMMARY OF OPCET VALUES IN S FOR THE DIFFERENT IIOT PROTOCOLS AND COMMUNICATION TECHNOLOGIES

IIoT protocol	ETH OPCET	5G-EC OPCET	Γ_{ETH-5G}
MQTT QoS0	3.95±0.01	4.30±0.09	0.35
OPC UA PubSub:10ms	4.32±0.05	4.67±0.15	0.35

OPCCLL performance between the ETH and 5G-EC versions of the MQTT QoS0 and OPC UA PubSub:10ms protocols, respectively. They are computed as the average of the difference in median OPCCLL performance between the cases considering 5G-EC and ETH technology for the 5 given movements of the robotic arm. For MQTT, the degradation in performance over 5G-EC as compared to ETH is, on average, 33.6 ms. In the OPC UA case, although the overall OPCCLL values are larger than for MQTT, the degradation when operating over 5G-EC as compared to ETH is slightly lower (22.2 ms). Considering both IIoT protocols, the operational control performance of the different movements executed by the robotic manipulator is, on average, 27.9 ms slower over 5G-EC than over ETH.

The results of the full operational cycle performance are summarized in Table IV, which includes OPCET mean values and standard deviations, considering both the MQTT QoS0 and OPC UA PubSub:10 ms IIoT protocols over both ETH and 5G-EC. The table also includes the parameter Γ_{ETH-5G} in ms, which assesses the difference in full manufacturing production cycle duration when operating over 5G-EC as compared to over ETH. It is computed as the difference between the average OPCET experienced over 5G-EC and the average OPCET experienced over ETH for the different protocols. As indicated, the considered MQTT implementation is more effective, as operational cycle times are shorter than with the OPC UA one. Over ETH, the average cycle time, considering the 10 movements of the robotic manipulator, is 3.95 s with MQTT, and 4.32 s with OPC UA. In the 5G-EC case, the cycle times over MQTT are also better than OPC UA by 0.37 s. The OPCET standard deviation gives an insight in terms of stability of the operational cycle time performance. MQTT over ETH provides the stablest performance with a low variability between cycle times of 0.01 s. This is slightly increased to 0.05 s in the OPC UA over ETH. Over 5G-EC, the variability increased further to 0.09 and 0.15 s for MQTT and OPC UA, respectively. To understand better this variability, in Figure 10, an OPCET dispersion diagram is given, where OPCET values are scatter-plotted against the average OPCET values for a given IIoT protocol and communication technology.

Overall, the impact of applying 5G-EC translates into a 0.35 s larger cycle time as compared to the reference ETH cases. This represents a 8.1-8.8% increase in cycle time by applying 5G-EC, which is due to the combined effect of all elements described along this paper: communication technology, computing capabilities, and robotic hardware capabilities an configuration settings. Although these 5G-

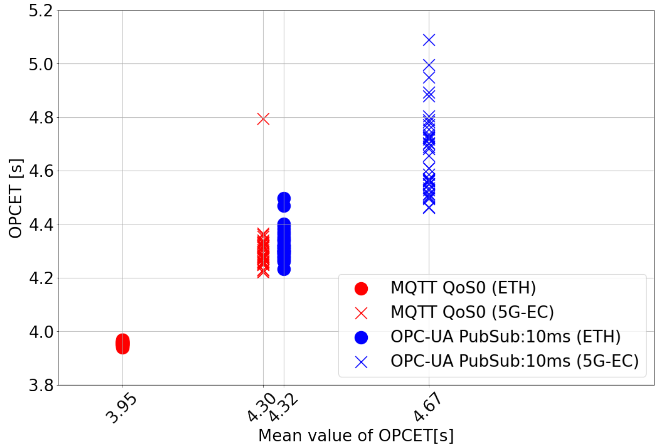

Fig. 10. Dispersion diagram of the OPCET for the different IIoT protocols and communication technologies in industrial operational conditions.

EC performance values might be perceived as negative at first glance, it should be reminded that 5G-EC enables that the operational robotic cell becomes wireless, which has a number of associated benefits in terms of flexibility, re-configurability, etc. Further, wireless robotic cells will diminish planned maintenance downtime as there will be no need to deal with re-cabling. Therefore, it is expected that operating robotic cells over 5G-EC will even result in positive production throughput gains when considering the full set of industrial production-related activities including planning, re-configuration, repairs and maintenance. It it also expected that with future 5G Releases (Rel.16 and above), the reported 5G-EC will improve, and get closer to that one over ETH when enabling specific ultra-reliable and low-latency features [22].

V. CONCLUSIONS

This paper presented a realistic wireless robotic cell environment operated over 5G and edge-cloud technologies considering two different IIoT communication protocols: MQTT and OPC UA. The performance evaluation in operational conditions considered key performance indicators in both the communication and manufacturing domains, indicating that, as compared to wired Ethernet, the wireless operation over 5G introduces 10-30 ms delays at communication level, which can add further to the 0.35 s when considering overall manufacturing cycle times including the combined effects of the communication, computing, and physical actions of the robotic manipulator. This translates into an 8-9% degradation in production throughput which should not be perceived as negative since the wireless robotic cell has other associated benefits such as the operational flexibility and re-configurability. In general, the results indicate that both protocols are functional, scalable, and exhibit a better performance with small payloads, with MQTT being better than OPC UA for this specific implementation.

Future work will focus on the integration and synchronization of multiple robotic manipulators over 5G, as well as the integration of new 5G Releases including low-latency features.

REFERENCES

[1] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0," *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 17–27, 2017.

[2] M. Lopez, S. B. Damsgaard *et al.*, "Towards the 5g-enabled factories of the future," in *2023 IEEE International Conference on Industrial Informatics (INDIN'23)*, 2023, pp. 01–08.

[3] I. Rodriguez, R. S. Mogensen *et al.*, "5g swarm production: Advanced industrial manufacturing concepts enabled by wireless automation," *IEEE Communications Magazine*, vol. 59, no. 1, pp. 48–54, 2021.

[4] B. Chen, J. Wan *et al.*, "Smart factory of industry 4.0: Key technologies, application case, and challenges," *IEEE Access*, vol. 6, pp. 6505–6519, 2018.

[5] J. Zhang and X. Fang, "Challenges and key technologies in robotic cell layout design and optimization," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 231, no. 15, pp. 2912–2924, 2017. [Online]. Available: https://doi.org/10.1177/0954406216642473

[6] D.-H. Luong, H.-T. Thieu *et al.*, "Cloudification and autoscaling orchestration for container-based mobile networks toward 5g: Experimentation, challenges and perspectives," in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1–7.

[7] X. Zhang, S. Lim *et al.*, "Integration of 5g and opc ua for smart manufacturing of the future," in *2023 IEEE/SICE International Symposium on System Integration (SII)*, 2023, pp. 1–6.

[8] S. Profanter, A. Tekat *et al.*, "Opc ua versus ros, dds, and mqtt: Performance evaluation of industry 4.0 protocols," in *2019 IEEE International Conference on Industrial Technology (ICIT)*, 2019, pp. 955–962.

[9] M. Silveira Rocha, G. Serpa Sestito *et al.*, "Performance comparison between opc ua and mqtt for data exchange," in *2018 Workshop on Metrology for Industry 4.0 and IoT*, 2018, pp. 175–179.

[10] MQTT version 3.1.1. [Online], Available: http://docs.oasis-open.org/mqtt/mqtt/v3 [Accessed Jul. 31, 2023].

[11] A. Stanford-Clark and H. L. Truong, "MQTT for Sensor Networks (MQTT-SN) protocol specification," *International business machines (IBM) Corporation version*, vol. 1, no. 2, pp. 1–28, 2013.

[12] S.-H. Leitner and W. Mahnke, "Opc ua - service-oriented architecture for industrial applications," *Softwaretechnik-Trends*, vol. 26, 2006.

[13] T. Hannelius, M. Salmenpera, and S. Kuikka, "Roadmap to adopting OPC UA," in *2008 6th IEEE International Conference on Industrial Informatics*. IEEE, 2008, pp. 756–761.

[14] AAU 5G Smart Production Lab. [Online], Available: https://www.5gsmartproduction.aau.dk/ [Accessed Jul. 31, 2023].

[15] UR5e - Lightweight, versatile cobot. [Online], Available: https://www.universal-robots.com/products/ur5-robot/ [Accessed Jul. 31, 2023].

[16] I. Rodriguez, R. S. Mogensen *et al.*, "An experimental framework for 5g wireless system integration into industry 4.0 applications," *Energies*, vol. 14, no. 15, 2021. [Online]. Available: https://www.mdpi.com/1996-1073/14/15/4444

[17] SIM8262E-M2 5G sub-6GHz. [Online], Available: https://techship.com/products/simcom-sim8262e-m2-5g-sub6/ [Accessed Jul. 31, 2023].

[18] S. B. Damsgaard, D. Segura *et al.*, "Commercial 5g npn and pn deployment options for industrial manufacturing: An empirical study of performance and complexity tradeoffs," in *2023 IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2023)*, 2023, pp. 01–07.

[19] M. Lopez, S. B. Damsgaard *et al.*, "Towards the 5G-Enabled Factories of the Future," in *2023 IEEE International Conference on Industrial Informatics*. IEEE, 2023.

[20] D. Arias-Cachero Rincon, "Kill-the-plc: Implementation and performance evaluation of a disruptive robotic cell environment based on 5g and edge cloud technologies," Master's thesis, University of Oviedo, 2023, [Online], Available: https://hdl.handle.net/10651/69235 [Accessed Jul. 31, 2023].

[21] iperf - the ultimate speed test tool for tcp, udp and sctp. [Online], Available: https://iperf.fr [Accessed Jul. 31, 2023].

[22] R. S. Mogensen, I. Rodriguez *et al.*, "Evaluation of the impact of wireless communication in production via factory digital twins," *Manufacturing Letters*, vol. 28, pp. 1–5, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2213846321000067