IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 21, 2021, accepted July 14, 2021, date of publication July 19, 2021, date of current version July 27, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3098109

Dissecting the Impact of Information and Communication Technologies on Digital Twins as a Service

MILAN GROSHEV[1], **CARLOS GUIMARÃES**[1], **ANTONIO DE LA OLIVA**[1], **(Member, IEEE), AND ROBERT GAZDA**[2], **(Member, IEEE)**
[1]Departamento de Ingenieria Telematica, Universidad Carlos III de Madrid, 28911 Madrid, Spain
[2]InterDigital Europe Ltd., Conshohocken, PA 19428, USA

Corresponding author: Milan Groshev (milan.groshev@uc3m.es)

This work was supported in part by the H2020 European Union/Taiwan (EU/TW) Joint Action 5G-eDge Intelligence for Vertical Experimentation (DIVE) under Grant 859881, in part by the H2020 5Growth Project under Grant 856709, in part by the Madrid Government (Comunidad de Madrid-Spain) through the Multiannual Agreement with Universidad Carlos III de Madrid (UC3M) in the line of Excellence of University Professors under Grant EPUC3M21, and in part by the context of the V PRICIT (Regional Program of Research and Technological Innovation).

ABSTRACT Recent advances on Edge computing, Network Function Virtualization (NFV) and 5G are stimulating the interest of the industrial sector to satisfy the stringent and real-time requirements of their applications. Digital Twin is a key piece in the industrial digital transformation and its benefits are very well studied in the literature. However, designing and implementing a Digital Twin system that integrates all the emerging technologies and meets the connectivity requirements (e.g., latency, reliability) is an ambitious task. Therefore, prototyping the system is required to gradually validate and optimize Digital Twin solutions. In this work, an Edge Robotics Digital Twin system is implemented as a prototype that embodies the concept of *Digital Twin as a Service* (DTaaS). Such system enables real-time applications such as visualization and remote control, requiring low-latency and high reliability. The capability of the system to offer potential savings by means of computation offloading are analyzed in different deployment configurations. Moreover, the impact of different wireless channels (e.g., 5G, 4G and WiFi) to support the data exchange between a physical device and its virtual components are assessed within operational Digital Twins. Results show that potentially 16% of CPU and 34% of MEM savings can be achieved by virtualizing and offloading software components in the Edge. In addition, they show that 5G connectivity enables remote control of 20 ms, appearing as the most promising radio access technology to support the main requirements of Digital Twin systems.

INDEX TERMS Digital Twin, Edge, NFV, 5G, distributed system, wireless networks.

I. INTRODUCTION

Industry 4.0, or the industry of the future, is offering a new way of understanding and organizing distinct manufacturing processes, pushed by the increasing interest of industrial verticals to accomplish a digital transformation of their industries. It focuses on an effective interaction between humans, machinery and products to build an efficient, agile and smart manufacturing environment. Intelligent production lines and flexible factory are the two of its key features that are contributing to meet the increased demands from customers [1], [2]i. Some countries (e.g., Japan, France, China, South Korea) already launched national plans of action to realize such evolution through the concept of a service-oriented industry [3].

A service-oriented industry shifts the implementation of the manufacturing processes, offering them as a service. A complex manufacturing task can be realized by a set of collaborative interoperable and platform independent services. As a consequence, capabilities like smart orchestration, dynamic scaling, and computation offloading become available. For example, a cloud-based service-oriented manufacturing enables resources and other services to be shared in a convenient pay-as-you-go manner [4]. Industrial verticals are

The associate editor coordinating the review of this manuscript and approving it for publication was Xujie Li.

then empowered with capabilities like dynamic reconfiguration, programability, on-demand usage, and sharing [5] aiming for minimal operational expenses, enhanced and faster production, and improvements on the time to market.

In this service-oriented concept the physical and virtual manufacturing processes are getting progressively interconnected, considered as the next-generation of smart systems. The Digital Twin arises as the key service to bridge both physical and virtual worlds through the integration of physical processes, computing and communications technologies [6]. It creates virtual objects in a virtual world that replicate the behaviour of the physical devices in the physical world through connected data [7]. Such environment paves the way for novel applications where the control of the manufacturing processes is performed in a real-time, remote and collaborative way, optimizing the whole manufacturing process [8].

In order to implement a service-oriented Digital Twin, advances on Information and Communication Technologies (ICT), such as Edge computing, Network Function Virtualization (NFV) and 5G, are playing a paramount role for satisfying the required Key Performance Indicators (KPIs) in terms of latency, reliability, bandwidth and scalability. Edge computing [9] offers low-latency, high bandwidth, network context-information and location awareness in Digital Twin services by providing computing capabilities at near proximity of the factory floor. Similar to Cloud computing, Edge computing is envisioned as a technology that relies on a virtualization environment and provides flexibility, scalability and easy management. Motivated by these benefits, ETSI has been first to create a standardized Edge environment through Multi-Access Edge Computing (MEC) [10], which is supported by the virtualization framework of NFV [11] that was also pioneered by ETSI. Lastly, besides the enhancements on the radio and spectrum usage, 5G is being natively designed to exploit different technology domains to e.g. implement network slicing over the same and shared infrastructure and provide logically isolated and dedicated end-to-end (E2E) networks for the sole use of the industrial verticals.

This research aims at analyzing and validating the integration of the aforementioned technologies to support the implementation of a Digital Twin as a Service (DTaaS). To do so, a Digital Twin service for robotic arms is split in virtualized functions in order to increase flexibility and reduce the burden on the physical devices. This work departs from the design of an Edge-based Digital Twin as an example of a DTaaS, incrementally validating the potential of such solution. First, through its mathematical formulation and, later, through a prototype implementation and a set of experiments performed over a 5G testbed. The implementation of the Edge Robotic Digital Twin service helps in narrowing down the scope of this work from general to specific and in quickly identifying the trade-offs. To the best of our knowledge, this is the first experimental evaluation that validates an Edge Robotic Digital Twin implemented as a DTaaS over 5G and Edge computing, analysing trade-offs regarding different deployment configurations. The main contributions of this work are the following:

- Analysis of the potential benefits of ICT to realize DTaaS;
- Implementation of a DTaaS prototype, namely an Edge Robotic Digital Twin featuring a robotic arm;
- Theoretical and experimental evaluation of the presented Edge Robotic Digital Twin;
- Elaboration on the computation offloading possibilities for Edge Robotic Digital Twin;
- Elaboration on the performance of potential wireless technologies to support the Edge Robotic Digital Twin;
- Identification of main lessons learnt and open issues for its implementation over a real industrial environment.

The rest of the paper is organized as follows. Section II introduces the concept of Digital Twin in Industry 4.0, analyses the characteristics of recent advances in ICT, and discusses the relevant related work. Section III provides an exemplary design and mathematical formulation of an Edge Robotic Digital Twin. Then, Section IV presents a prototype implementation of the designed Edge Robotic Digital Twin, including baseline performance observations. The implemented prototype is then used to study the main candidates for VNF offloading in Section V, and the impact of different Radio Access Technologies (RATs) in Section VI. Section VII discusses the main insights from the obtained results. Finally, concluding remarks and future directions are given in Section VIII.

II. DIGITAL TWIN SYSTEMS FOR INDUSTRY 4.0

Digital Twin in manufacturing integrates any industrial process achieved through closed-loop feedback mechanisms. The virtual factory includes geometrical and virtual models of tools, machines, operatives, finished products, etc., as well as behaviors, rules, physics and analytic models. As such, based on their functionality, the Digital Twin can be classified into: *(i)* Monitoring; *(ii)* Simulation; and *(iii)* Operational Digital Twins. While the Monitoring Digital Twin provides information about the operational states or behaviors of the physical device (e.g., production line dashboard), the Simulation Digital Twin contains 3D models, simulation tools and machine learning models to describe, understand and predict the future behaviour of the physical device. Lastly, the Operational Digital Twin enables the manufacturing worker to interact with the physical device and execute different recommended actions (e.g, remote control) to enhance the industrial process.

Motivated by the advent of Industry 4.0, different standardization bodies and alliances, such as ETSI, 3GPP, 5G Alliance for Connected Industries and Automation (5G ACIA) and Next Generation Mobile Networks Alliance (NGMN), defined new use cases with distinct connectivity requirements that connect people, objects, processes and systems [12]–[15]. Table 1 summarizes the key connectivity requirements of different industrial use cases mapped into the Digital Twin categorization. For example, Monitoring Digital

TABLE 1. Connectivity requirements of Digital Twin.

Digital Twin	Latency	Data Rate	Reliability	Scalability
Monitoring	50-100 ms	0.1-0.5 Mbps	99.9%	100-1000 nodes
Simulation	20-50 ms	1-1000 Mbps	99.99%	1-100 nodes
Operation	0.5-20 ms	1-100 Mbps	99.9999%	1-50 nodes

Twins can be applied to use cases such as industrial condition monitoring and process automation based on sensors. Simulation Digital Twins can be applied to factory remote maintenance, whereas the Operational Digital Twins includes remote operation, motion control, safe control and closed-loop control use cases.

A. ENABLERS FOR INDUSTRY 4.0-BASED DIGITAL TWIN

Digital Twin solutions in manufacturing need to integrate network virtualization, computing and wireless technologies, so that the most stringent requirements of real-time applications are met.

1) DIGITAL TWIN AS A SERVICE

Virtualized Network Function (VNFs), also referred to as service functions, are widely embedded in network operators' deployments. It allows them to migrate network functionalities from costly vendor hardware to general-purpose resources. Network Services are usually formed as a composition of VNFs, each providing a specific functionality of the whole service. In addition, functional network modules that compose the Network Service are split, centralizing a subset of functions into a data center. This functional split was proposed as a feature for the next-generation of Radio Access Network (RAN), where the processing of all the base stations has been centralized into a data center. Pooling computational resources reduces the cost of 5G deployments and centralization enables easier coordination between next-generation base stations [16].

The adaptation of industrial processes towards a concept based on Digital Twin is facing a major challenge with respect to the unification and exchange of information by different applications [17]. To make use of the Digital Twin, applications must have access to information from different systems, making abstractions a very important topic to handle the underlying heterogeneity. Moreover, manufacturing service encapsulation, composition and publication are identified as key technologies that need to be studied [18].

Applying the Network Service and functional split concepts to Digital Twin gives the factory operator a unified and modular view of the system and extends the management capabilities beyond the networking aspects to implement a Digital Twin as a Service. These concepts enhance the flexibility of the infrastructure as well as of monitoring and management operations, contributing to fulfil the scalability and availability requirements.

2) COMPUTATION OFFLOADING FOR DIGITAL TWIN

Digital Twin systems represented as a Network Service can be deployed at the Cloud. Cloud-based Digital Twins are suitable for delay tolerant, long-term, high computational applications (e.g., monitoring or simulation Digital Twins) crucial for preventive maintenance and business-decision support. However, proper execution of a Digital Twin in manufacturing needs real-time access to production data, making real-time connection and synchronization a major challenge. In fact, the integration of the physical and virtual worlds of manufacturing is very difficult due to the variable and uncertain characteristics of the factory floor [19]. Moreover, the manufacturing sector, as the largest world energy consumer (55% of world's energy [20]) is in a constant search for energy efficient optimization strategies that will reduce the energy consumption.

Edge computing offers performance and cost related benefits for Digital Twin solutions by moving the Cloud concepts to the edge of the network. Consequently, it contributes to a reduction on the E2E latency, increases bandwidth savings in the transport network, and improves availability of the whole service. It reduces the cost of physical devices by offloading computation modules in an Edge server. Still, the connected data can be processed locally in the Edge without extra jitter or packet loss caused by the communication towards the Cloud [21]. By facilitating local and real-time decisions, the accuracy of control processes in operational Digital Twin is improved with minimum cost. In addition, Edge-based Digital Twin leverages localization and network contextual information in order to achieve optimal usage of physical devices, operation and maintenance tasks, and energy consumption reductions. Applied in industrial environments, Edge computing provides distributed and controlled environment that can significantly reduce the overall energy consumption and improve the real-time data feedback within the factory floor.

3) 5G FOR DIGITAL TWIN

Wireless technologies provide a number of benefits for manufacturing, such as greater flexibility for connecting physical devices, reductions of installation and maintenance costs, support for mobility and improved safety for the employees [22]. As of today, wired networks are widely used in the industry automation due to the fact that wireless technologies fall short on meeting the stringent requirements of real-time applications. Consequently, the physical devices are connected through wired technologies such as Fieldbus [23] and industrial Ethernet [24], which are costly and inflexible.

With the emergence of 5G, Industry 4.0 recognized the opportunity for a unified communication interface that can support the requirements of any type of Digital Twin through three main service profiles: *(i) enhanced Mobile Broad-Band* (eMBB); *(ii) massive Machine-Type Communication (mMTC)*; and *(iii) Ultra Reliable Low-Latency Communications* (URLLC). 5G provides guaranteed Quality of Service (QoS) for time critical applications as a built-in feature.

In addition, 5G also defines support for handling mobility in industrial environments that opens the possibility to apply the Digital Twin concept for movable platforms (e.g., mobile robots, automated guided vehicles). Finally, network softwarization and virtualization are natively explored by 5G to support network slicing capabilities, allowing the creation of logically isolated and dedicated networks over the same and shared infrastructure tied to the specific requirements of each service and/or application.

B. APPLICATIONS IN REAL INDUSTRIAL ENVIRONMENTS

The world of manufacturing is always looking for a way to improve time to market and save money. It is also an area that has many sub-fields, from large factories to small machines and sensors. For this reason, the majority of existing research studies revolve around the main applications for Digital Twins in different manufacturing sectors, arguing on the importance of emerging technologies for the development of smart manufacturing [7], [19], [25], [26]. Digital Twins are applied in the automotive industry to monitor the production of automobiles, making punctual modifications, and optimize processes [27]. Moreover, they facilitate the adjustment of the industrial production operations through autonomous factory planning [28], [29] as well as the lifecycle management of the production phase in small and medium-sized enterprises (SME) [30]–[32]. Lastly, Digital Twins also serve as a simulation-based environment that help realize complex control algorithms for real-time production control [33].

Industrial Cloud [34] and Edge computing [21] are known concepts to the manufacturing world. They enable computation and control to be offloaded to a computing infrastructure, making Digital Twins more scalable, real-time and ensure accessibility to the physical devices anytime and from anywhere. Currently, Cloud-based Digital Twin frameworks for industrial robotics [35], manufacturing execution systems [36], [37] and smart product-service systems [38] have been developed to provide an insight into smart manufacturing. Edge-based Digital Twins have also been studied in the literature with their benefits for manufacturing such as anomaly detection for automation systems [39] and improving production quality, efficiency and costs for metal additive manufacturing system [40].

Another key driving feature of Digital Twins in the automation and manufacturing industries is industrial wireless networks. The potential advantages from wireless technologies are significant and they have been studied through papers that elaborate on the importance of the industrial wireless [22] and 5G environment [41]–[43] for smart manufacturing. Beside conceptual works, some recent study investigates Digital Twin solutions over IoT communication technologies (LTE-M, LoraWan and Sigfox) for smart manufacturing assembly systems [44]. A Digital Twin prototype for mission critical application over 4G was developed to identify the main obstacles and cyber-security issues in the realization of an Industry 4.0 vision [45].

C. MIND THE GAP

The potential advantages from ICT for Digital Twins in the manufacturing sector are significant, but realizing them remains a challenging task. Despite all the expectations that are studied in the literature, industrial verticals remain skeptical on the maturity of recent advances in ICT. There is a lack of experimental studies that deal with integration of these ICT with current network deployments. This is a continuous process in which validation and optimization are crucial to fully deploy Digital Twin systems. Consequently, this study presents a Operational Digital Twin prototype that builds on top of 5G connectivity, Edge computing and NFV, aiming to give insights into the potential returns and performance improvements for industrial verticals willing to invest on 5G and Edge deployments.

III. EDGE ROBOTIC DIGITAL TWIN

In this work, an Edge Robotic Digital Twin is developed as an implementation of an Operational Digital Twin to be provided as a service. It envisages a set of robotic arms that are remotely controlled, coordinated and monitored to perform different tasks in a factory floor comprising multiple RATs. Such an example is foreseen in the manufacturing industry, where existing robotic arms are enhanced with Digital Twins capabilities and used to optimize the production line by e.g. eliminating defective pieces and repetitively performing accurate tasks at high-speed.

This scenario opens up new business roles and the introduction of new stakeholders in the value chain. The owner of the computing and/or communication infrastructure (e.g., micro-datacenter and/or 5G network) can be the industrial vertical itself or provided by a third party infrastructure provider that deploys and manages it for the industrial vertical in the form of a service. In turn, the Edge Robotic Digital Twin service can be provided by a robotics application provider, which delivers its applications through the underlying infrastructure and establishes the connection to the robotic systems.

A. EDGE ROBOTIC DIGITAL TWIN SERVICE

Figure 1 illustrates the main components that comprise the Edge Robotic Digital Twin service. The infrastructure is composed by an on-premises or off-premises Edge server providing the computing and storage capabilities, and a 5G network providing the communication capabilities within the factory floor. The *Robot Drivers* act only as sensors and actuators (i.e., bare metal devices with minimal resources), while the Digital Twin and the *Robot Stack* implement the control, navigating and monitoring operations on top of the robotic systems. The *Robot Drivers* always remains in the robotic system but the *Robot Stack* can be distributed across the robot or an Edge server. Accordingly, the Edge Robotics Digital Twin service implemented for robotic arms is decomposed into five VNFs, described as follows:

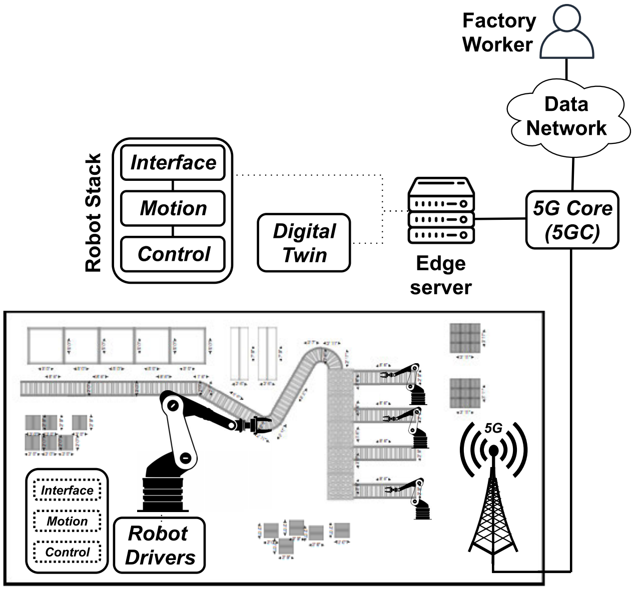


FIGURE 1. Edge robotic Digital Twin service.

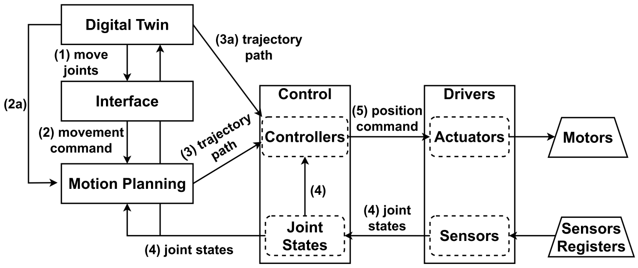


FIGURE 2. Sequence message diagram for remote control.

- The *Robot Drivers VNF* are low-level software interfaces that control and monitor the specific robotic arm hardware. Therefore, the *Robot Drivers VNF* interacts directly with the robot hardware and is responsible for (*i*) executing in robotic arm's hardware of any command received from the *Control VNF*; and (*ii*) making available sensor data and operational states to the *Control VNF*.
- The *Control VNF* implements generic control-loop feedback mechanisms used for robot manipulation. It takes a trajectory, composed by position commands, as input and runs a control-loop, following a given frequency, towards the *Robot Drivers VNF*. This VNF defines a set of controllers that are used by the remaining VNFs for robot manipulation.
- The *Motion Planning VNF* receives a movement command (e.g., go to position) and computes the trajectory for the robotic arm. The created trajectory consists of path commands, being sent to the *Control VNF* in order to move the robotic arm from its current position to the desired one.
- The *Interface VNF* is a high-level abstraction for the core robot functionalities. The main goal of this VNF is to hide the complexity by providing the interaction through high-level commands. It acts as the gateway between the user application and the robotic arm.
- The *Digital Twin VNF* gives human understanding of the connected data by implementing 3D, control and analytic models. While the 3D models provide a visual representation of the variation of the robotic arm, the control models enables remote control and maintenance. Finally, the analytic models define descriptive (e.g, remote monitoring), diagnostic (e.g., understand what happens) and predictive (e.g., when it will happen again and how to respond) behavior of the robotic arm.

Figure 2 depicts the interactions between the different VNFs of the Edge Robotics Digital Twin service. When a user needs to remotely control a robotic arm, it issues a **move joints** (step 1) manipulation command using the selected interfacing device (e.g., joystick, AR/VR). The move joints command is sent to the *Interface VNF* which offers a custom-made interface (e.g., Python or REST API), translating it in a robot specific movement command. Then, the *Interface VNF* sends the **movement command** (step 2) to the *Motion Planning VNF*. When the movement command is received, the *Motion Planning VNF* performs several command validations and generates the **trajectory path** consisting of an array of position commands. For each position command, each joint is given a specific position, velocity and acceleration. Once the *Control VNF* receives the **trajectory path** (step 3), it runs a control-loop against the *Robot Drivers VNF*. The control-loop starts with the *Robot Drivers VNF* sending the **joint states** (step 4) of the robotic arm to the *Control VNF*. This information is also propagated to the *Digital Twin* in order to update the virtual model. Next, the *Control VNF* interpolates the received trajectory path to get the next position command. The control-loop is then closed when the *Control VNF* sends the **position command** (step 5) to the *Robot Drivers VNF*. Note that the *Digital Twin* can also be configured to send actions to the robotic arm directly via the *Motion planning* or *Control VNFs* (steps 2a and 3a).

B. MATHEMATICAL FORMULATION

As a first validation stage, a simplistic mathematical formulation of the Edge Robotic Digital Twin is modeled and implemented as a network service composed of different VNFs distributed across the robotic arm and the Edge or Cloud. In this formulation, different wireless technologies are considered to provide the connectivity between the robotic arm and the network, namely 4G, 5G and WiFi. 4G and 5G are the current and future wireless technology in the licensed spectrum, whereas WiFi is the most widespread wireless technology in the unlicensed spectrum. The Edge is considered to be on-path, thus collocated with the RAN, while the Cloud is deployed in geographical location disjoint from the path between the factory floor and the manufacturing worker.

TABLE 2. Notation Table.

Feature	Name	Simulation Values
$d(G_s)$	E2E latency for service graph	–
$d_p(G_s)$	computing delay for service graph	–
$d_t(w)$	propagation delay (wireless network)	–
$d_t(t)$	propagation delay (transport network)	–
G_s	service graph	PNFs: 1 VNFs: 4
$d_p(v_t)$	processing delay of a VNF	Robotic Arm: 2 ms Edge: 1 ms Cloud: 0.5 ms
δ_w	baseline delay (wireless network)	5G (eMBB): 5 ms 5G (URLLC): 1ms 4G: 15 ms Wifi: 2 ms
β_w	deviation factor (wireless network)	5G: 1.30 4G: 1.30 Wifi: 2 (baseline), . 10 (jitter)
r_w	reliability penalty factor (wireless network)	5G: 1.00001 4G: 1.001 WiFi: 1 (baseline), . 1.2 (loss)
δ_t	delay per distance unit (transport network)	0.5ms per 100 km
β_t	deviation factor (transport channel)	1
α_t	E2E distance (transport network)	Factory Worker: 100 km Cloud: 1000 km

Table 2 presents a summary of the notation used in this mathematical formulation, along with the simulation values.

The network service $s \in S$ is described through a service graph G_s composed of sequential VNFs $v \in N(G_s)$. While upstream flows are processed first by the VNFs $v \in V_r \subset N(G_s)$ deployed in the robotic arm and later by the VNFs $v \in V_e \subset N(G_s)$ deployed in the edge, downstream flows follow the reverse order. The computing requirements of each VNF $v \in N(G_s)$ is defined by the set of tasks $\{v_t\}_{t=0}^{T_v}$ they need to compute. As such, the computing requirements of both the robotic arm, Edge and Cloud have associated a time to compute each of their tasks v_t. In other words, $d_p(v_t)$ defines the processing delay of a VNF task v_t. The total processing delay of a network service is defined as:

$$d_p(G_s) = \sum_{v \in N(G_s)} \sum_{t=0}^{T_v} d_p(v_t) \tag{1}$$

The connection of the robotic arm to the RAN is established through wireless technologies $\mathcal{W}$, more specifically, WiFi, 4G (LTE), and 5G (5G-NR). Each of them have an associated propagation time, and a deviation probability $\beta_w \geq 1,\ w \in \mathcal{W}$. In addition, a penalty factor $r_w \geq 1,\ w \in \mathcal{W}$ is considered for each technology that allows reliability (i.e., retransmissions) to be defined as a time component. Hence, the propagation time for each wireless technology is defined as follows:

$$d_t(w) = (\delta_w \cdot \beta_w) \cdot r_w, \quad w \in \mathcal{W} \tag{2}$$

with δ_w denoting the baseline delay for the wireless technology w.

Finally, the path from the RAN up to the Edge and/or Digital Twin application (i.e., the transport network) consumes a delay $d_t(t)$ related to the propagation along the fixed and wired links $\mathcal{T}$, and a deviation probability $\beta_t \geq 1,\ t \in \mathcal{T}$. In addition, a penalty factor $r_t \geq 1,\ t \in \mathcal{T}$ is considered that allows reliability (i.e., retransmissions) to be defined as a time component. Hence, the propagation time in the transport network is defined as follows:

$$d_t(t) = (\delta_t \cdot \beta_t) \cdot \alpha_t, \quad t \in \mathcal{T} \tag{3}$$

Summing up, the E2E latency between the robotic arm and its digital twin is defined as follows:

$$d(G_s) = d_p(G_s) + \sum_{w \in \mathcal{W}} d_t(w) \cdot u(w, G_s) + d_t(t) \tag{4}$$

with $u(w, G_s) \in \{0, 1\}$ denoting if the service graph G_s uses the wireless technology w. Note that we impose $\sum_{w \in \mathcal{W}} u(w, G_s) = 1$ for every network service s, i.e., we assume a single wireless technology is used.

1) SIMULATION ANALYSIS

Results from the implementation of the abovementioned model are presented in Figure 3 and Figure 4, in the form of Cumulative Distribution Function (CDF). The former presents the results of different deployment configurations by taking into account theoretical values for both the computing technologies and the wireless links. In turn, the latter shows the performance of a WiFi link when in presence of interference, which result in increased packet loss and jitter.

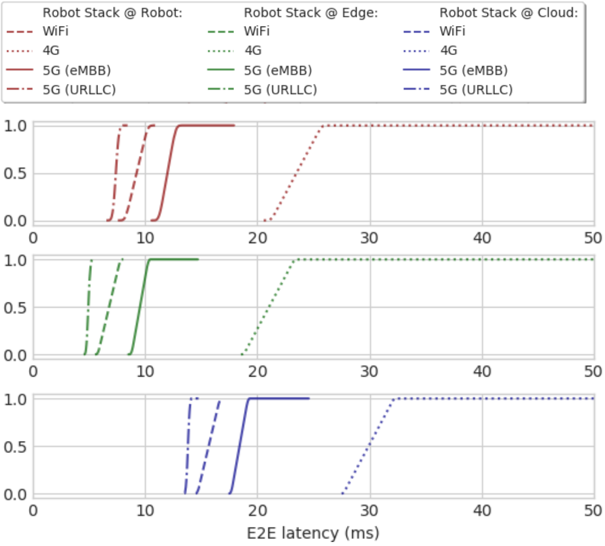


FIGURE 3. Different deployment configurations.

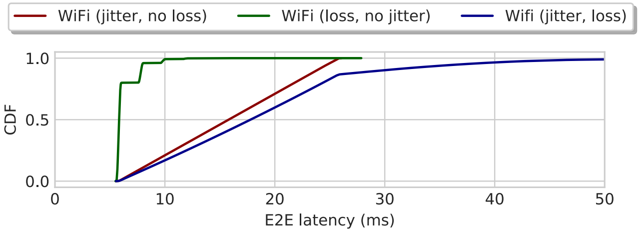


FIGURE 4. **WiFi with interference.**

The following insights can be extracted from this theoretical analysis:

- Radio Access Technology segment:
  - WiFi (in optimal conditions) is the best wireless technology (dashed lines in Figure 3). However, due to likely interference as it operates on an unlicensed spectrum, WiFi is not a reliable wireless channel for Digital Twins (Figure 4).
  - 4G cannot accommodate the minimum requirements in terms of latency in most situations (dotted lines in Figure 3).
  - 5G arises as the best wireless channel able to meet the latency requirements (solid and dashed-dot lines in Figure 3) with the least likelihood of interference.
- Computing technologies segment:
  - Moving the *Robot Stack* to Edge enables improvements in the RTT latency since tasks are computed faster than in the robotic arm (green lines in Figure 3 have a slight improvement when compared to the red lines).
  - Moving the *Robot Stack* to Cloud enables even more improvements in the computation speed but the longer distance to reach the cloud nullified the added benefits in terms of RTT latency (blue lines in Figure 3).
- Data network segment:
  - The maximum distance of between the physical object and the virtual replicas is tightly related to the RTT latency requirements of the Digital Twin application (green lines in Figure 3 when compared to blue lines).

The previous results provided hints for narrowing down the experimental study presented in the following sections: *(i)* on-device computing or Edge-offloading as the feasible approaches; and *(ii)* 5G as the most promising wireless channel, although WiFi and 4G are still considered for comparison purposes.

IV. PROTOTYPE IMPLEMENTATION AND BASELINE PERFORMANCE

To better experimentally evaluate the Edge Robotic Digital Twin, a prototype is implemented and deployed as a service in 5TONIC[1] laboratory in Madrid (Spain).[2] The first objective is to verify the proposed deployment and to assess the baseline performance of each component. The second objective is an investigation of the impact that component offloading has in the synchronization between the robotic arm and its Digital Twin. Lastly, the third objective is to assess the potential resource savings and the impact of different wireless technologies in the operation of the Edge Robotic Digital Twin.

The prototype presented in Figure 5 is composed of 6-axis Niryo One robotic arm that is connected to an Edge server with 8GB RAM and 4vCPU@2.4GHz via Customer Premises Equipment (CPE). Therefore, the communication between the robot and the Edge server comprises of wireless link, from the CPE to the referent RAT, and a wired network,

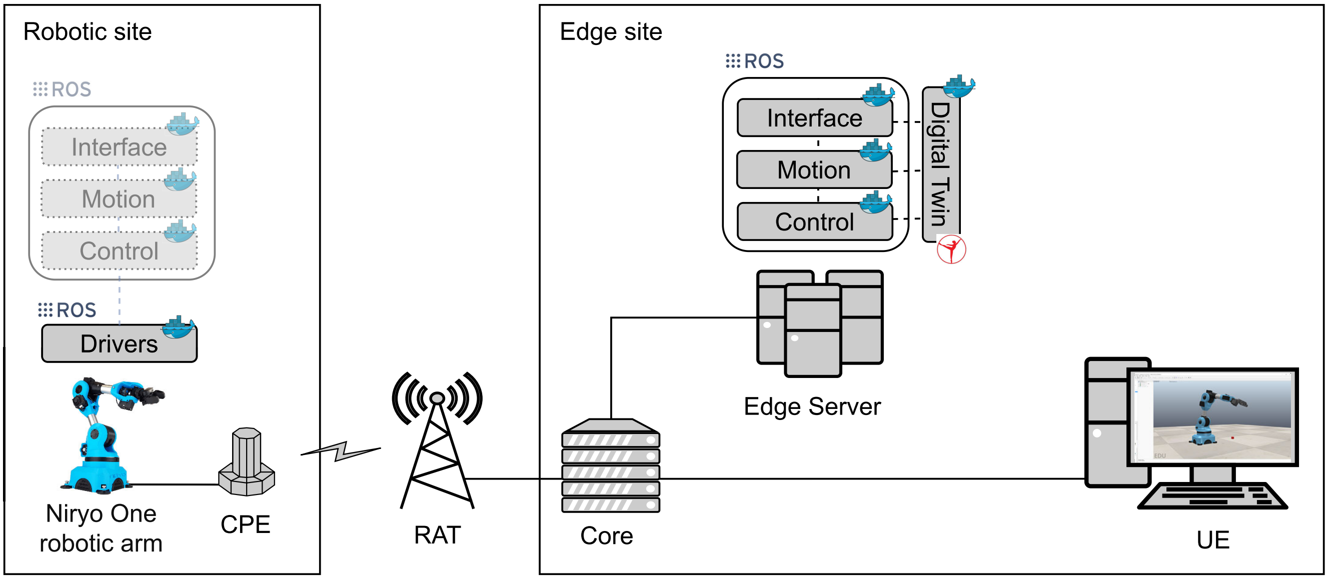


FIGURE 5. **Edge-based Digital Twin prototype implementation.**

---

[1] https://www.5tonic.org

[2] Demonstration video: https://youtu.be/b9jme58dLt8.

from the RAT to the Edge server. The Niryo One robotic arm[3] is equipped with Raspberry Pi 3 with 1.2 GHz 64-bit CPU and 1GB RAM. The robot maximum speed is 0.4 m/s for the steeper axes and 90°/s for the servo axis with configured ROS control frequency of 50Hz (i.e., 20 ms). The sampling frequency for reading the sensors registers (joint states) is configured at 50Hz (i.e., 20 ms). The control and the sampling frequencies are two independent and non-alignment open-loops. The Niryo One is open source collaborative robot designed for research in Robotics and Industry 4.0. It has similar design and functionalities as 6-axis industrial robot. In addition, the low latency requirements of the Operational Digital Twin can be fulfilled with Niryo One robotic arm which makes it a good candidate for this study. Although Niryo One software stack is based on Robot Operating System version 1 (ROS-1) as provided by the robot manufacture,[4] this work splits ROS-1 packages composing the Niryo One stack in several VNFs that are compatible with the developed Edge Robotic Digital Twin from Section III-A. Moreover, VNFs are packaged either as virtual machines or containers.

The different VNFs created from the provided Niryo One stack (i.e., *Robot Drivers*, *Control*, *Motion Planning* and *Interface VNFs*) were implemented as a Docker container. The *Robot Drivers VNF* are deployed in the robotic arm, while the remaining VNFs of the *Robot Stack* are distributed across the robotic arm or the Edge server envisioning different deployment configurations.

The Digital Twin is deployed in the Edge server as Docker container and implemented using Coppelia SIM[5] robot simulator, and providing a remote GUI for the factory worker. Coppelia SIM is selected due to its support for the Niryo One virtual replica and ROS-based communications. As such, two LUA scripts are implemented on top of Coppelia SIM: one to synchronize the robotic arm and its Digital Twin, and another to remotely control the robotic arm.

Each VNF of the implemented prototype is following the interactions presented in Figure 2 and described in Section III-A. Since ROS-1 is used as the robotic middleware, each VNF of this prototype is composed of one or more software components called ROS nodes. ROS provides a publish-subscribe messaging framework over which nodes exchange messages. By connecting to the ROS master, nodes register and locate each other. Once registered, nodes exchange messages via configurable topics in a peer-to-peer fashion over TCP.

A. BASELINE PERFORMANCE

Although this prototype is envisioning RATs for connectivity between the robot and the network, in order to obtain most reliable baseline values for later comparison with wireless technologies (Section VI), the Niryo One robot is connected to the Edge server using Ethernet connection. Ethernet as a technology is proven that can fulfil the Edge Robotics Digital Twin requirements. Two deployment configurations are considered: *(i)* the *Control*, *Motion Planning* and *Interface* VNFs deployed in the robotic arm; and *(ii)* the *Control*, *Motion Planning* and *Interface* VNFs offloaded in the Edge server. In both experiments, the Digital twin interacts with different VNFs from the *Robot Stack*, remotely controlling 1-axis of the robotic arm controlled for a period of 30 seconds and a movement offset of 0.010rad.

Table 3 presents the closed-loop when interacting with different VNFs from the *Robot Stack*. The closed-loop can be seen as the Digital Twin user experience and is defined as the time elapsed from the moment a remote command is issued from the Digital Twin until the respective robot joint states are received back. From the average values, the *Interface* and *Motion Planning VNFs* offer slower closed-loop (∼498 ms and ∼388 ms, respectively) with respect to the *Control VNF* (∼23 ms). This comes natural as the *Interface* and *Motion Planning VNFs* represent the transition between the physical and virtual space. To do so, they implement a set of translations and validations that require additional processing for each command. In turn, the *Control VNF* offers near real-time control by performing straight-forward operations. Regarding the processing delay, the *Control VNF* results in fastest processing because there are no command validations implemented, thus it may occur that a given command is enforced in the robotic arm even if it is not achievable. Moreover, it leaves the trajectory computation to the application developer, unlike the *Interface* and *Motion Planning VNFs* which offer optimal path computation. The optimal path computation and validation comes at a price of processing delay, where the *Motion Planning VNF* needs ∼365 ms and the *Interface VNF* 112.7 ms to compute the trajectory for a simple command.

TABLE 3. Baseline performance of interfacing at different layers.

Interfacing Layer		Closed-loop (ms)	Processing delay (ms)
Interface	Robot	497.976 ± 5.023	109.723 ± 1.106
	Edge	409.535 ± 0.896	66.284 ± 0.145
Motion Planning	Robot	388.253 ± 2.365	365.178 ± 2.224
	Edge	343.251 ± 4.291	320.199 ± 4.002
Control	Robot	23.075 ± 0.005	3.075 ± 0.001
	Edge	23.052 ± 0.003	3.052 ± 0.001

Figure 6 illustrates the position of the robotic arm and its virtual replica over time (bottom) when interfacing with the Control VNF (approximately 23ms closed-loop). Both positions are used to compute the synchronization accuracy over time (top). Results show that it is feasible to offload the *Robot Stack* to an Edge server and still maintain similar synchronization levels between the physical and virtual replica (97.72±0.65% and 97.66±0.67%). This desynchronization is due to the non-alignment of the control and sampling frequencies, plus the time required to physically move all joints in the robotic arm. Moreover, Table 3 shows that there

---

[3] https://niryo.com/product/niryo-one

[4] Widespread framework for developing and testing multi-vendor robotics software.

[5] https://www.coppeliarobotics.com

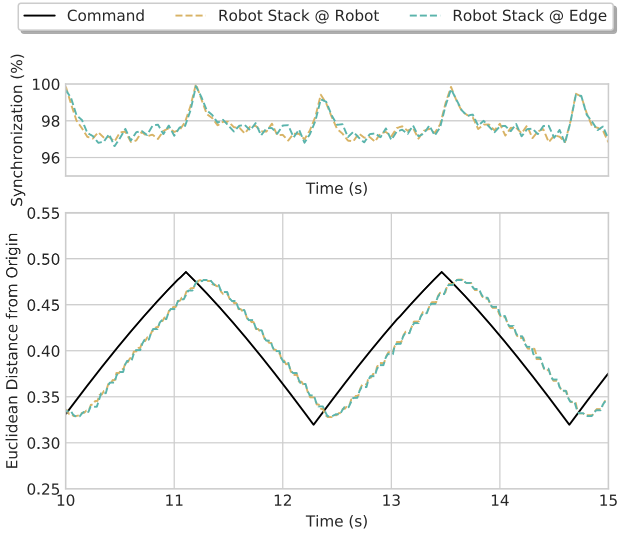


FIGURE 6. **Digital Twin performance comparison.**

is a speed up on the closed-loop for the *Interface* and *Motion Planning VNF* when the *Robot Stack* is offloaded to the more powerful Edge server. Such behaviour is related to faster computation capabilities of the Edge server with respect to the limited computation capabilities at the robotic arm. At the *Control VNF* such decrease is not appreciated due to the low computational needs.

V. RESOURCE CONSUMPTION AND SAVINGS POTENTIAL

To look into the scale of the on-device resource consumption and potential offloading savings, several deployment configurations are considered in this section for the implemented Edge Robotics Digital Twin. This analysis aims to identify the main candidates for VNF offloading by assessing the potential resource savings in terms of CPU, MEM, network usage, and energy consumption. It is worth mentioning that the experimental evaluation in this section is performed over wired technology. The main reason is because when investigating the potential on-device resource savings in our prototype, the bottleneck is not on the communication infrastructure.

The Niryo One robotic arm is connected to the Digital Twin using 10Gb Ethernet connection. Each experimental run consists of the Digital Twin controlling 1-axis of the robotic arm for 30 seconds and a movement offset of 0.010rad. The *Interface VNF* is used as the interfacing VNF to force the whole *Robot Stack* to process every command. Experiments consider the following on-device configurations: *(i) Robot Drivers VNF*; *(ii) Robot Drivers* and *Control VNFs*; *(iii) Robot Drivers*, *Control* and *Motion Planning VNFs*; and *(iv)* full *Robot Stack*. The described experiments are repeated 10 times for each deployment, being presented the average values and their standard deviation.

Figure 7 presents the average resources usage in terms of CPU, MEM, network and energy consumption for all the deployment configurations while processing navigation commands. The obtained data for the CPU, MEM and network consumption is measured in the robotic arm with a python script using the cross-platform library psutil.[6] The energy consumption data is measured with an energy consumption meter installed in the power supply of the robotic arm.

It can be seen from Figure 7a that the CPU consumption of the whole *Robot Stack* is approximately 30% of the total CPU consumption on the robotic arm. Excluding the *Robot Drivers VNF* that must be deployed in the robotic arm, the CPU consumption can be potentially reduced by around 16% by offloading all the remaining VNFs. In particular, around 10% by offloading the *Interface VNF*, around 5% by offloading the *Motion Planning VNF*, and around 1% by offloading the *Control VNF*. The *Interface VNF* has higher computational requirements, needed to perform the command translation, validation and real-time synchronization between the Digital Twin and the ROS systems. For what concerns the Control and *Motion Planning VNFs*, the CPU consumption of the *Control VNF* is mainly due to the straightforward control-loop mechanism that interpolates and forwards position commands, while in the *Motion Planning VNF* this is a result of a simple path planning in our experiment runs.

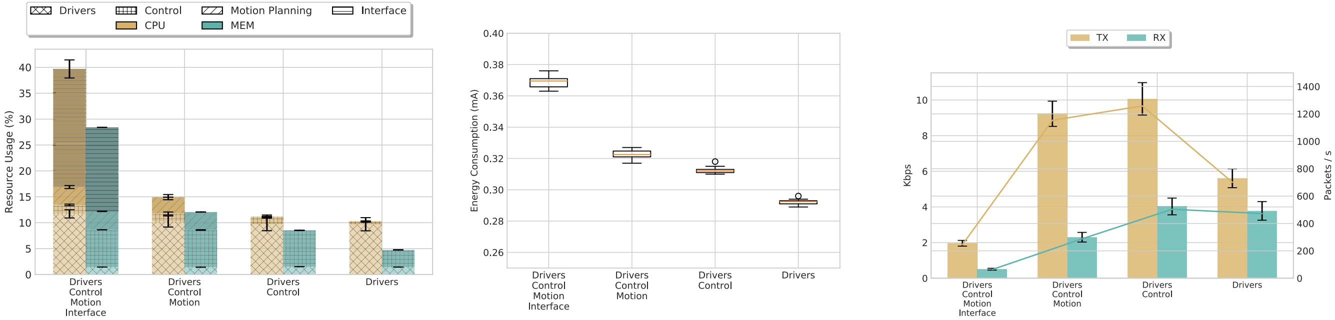
Regarding MEM usage, the total MEM consumption on the robotic arm is approximately 50% as shown in Figure 7a. This reflects 16% used by the *Robot Drivers VNF*, meaning that around 34% MEM savings can be achieved by offloading the remaining VNFs. Similarly, the *Interface VNF* offers highest potential savings of around 23% of the total MEM usage, while the *Motion Planning* and *Control VNFs* around 7% and 4%, respectively. The 23% of MEM used by the *Interface VNF* is partially due to the robot action server that handles Digital Twin concurrent requests, checks if the command can be processed, validates parameters and calls required controllers (e.g., *Motion Planning VNF* or *Control VNF*). Additionally, for the *Motion Planning* and *Control VNFs*, the MEM usage is significantly smaller since the robotic arm receives simplified navigation commands from the *Interface VNF*.

The energy consumption (Figure 7b) follows the CPU and MEM usage pattern from Figure 7a, where most energy is consumed when the *Robot Stack* is deployed in the robotic arm. As VNFs are offloaded to the Edge, the energy consumption in the robotic arm decreases, proving that energy savings in devices can be achieved by offloading components in the Edge. By moving the energy contribution mostly to the data centers at the Edge, different energy-aware optimization models are easier to implement [46].

Figure 7c shows that a deployment configuration where the whole *Robot Stack* is in the robotic arm introduces the smallest overhead in terms of network utilization. This comes naturally because in this deployment configuration the robotic arm transmits (TX) only the joint states to the Digital Twin and receives (RX) high-level commands from

---

[6] https://pypi.org/project/psutil



(a) CPU and MEM (b) Energy Consumption (c) TX and RX

FIGURE 7. Resource consumption on robotic arm.

the Digital Twin processed by the *Interface VNF* at lower frequency. Results also show that from all deployment configurations, deploying only the *Robot Drivers VNF* in the robotic arm introduces less overhead into the communication link. However, there is still an increment of the network utilization compared to a self-contained deployment configuration in the robotic arm. Such behaviour is mainly due to the offloaded *Control VNF*. While in upstream the *Robot Drivers VNF* needs to transmit the joint states to the *Control VNF*, in downstream this increment is due to the higher frequency of commands sent by the *Control VNF* to the robotic arm. In the case when the *Control VNF* is deployed in the robotic arm a significant increase of TX is witnessed. This is mainly because the *Control VNF* needs to continuously interact with the *Interface VNF* in order to enable robot manipulation.

The resource consumption evaluated in the experiments above clearly showed that the *Interface* and *Motion Planning VNFs* are the most likely targets for potential resource savings. However, when considering the network overhead that the *Control VNF* introduces on the communication link, its offloading should be also considered.

VI. IMPACT OF RADIO ACCESS TECHNOLOGIES

An Edge Robotic Digital Twin system places stringent networking requirements that must be fulfilled by the underlying RAT. In this section, the applicability of currently available RATs to support the proposed design is analysed, together with their impact on the overall performance of the Edge Robotic Digital Twin service. To do so, the Niryo One robotic arm is connected to the Digital twin using WiFi, 4G and 5G. WiFi connectivity relies on IEEE 802.11ac compatible devices working on the 2.4Gz band, while for 4G and 5G connectivity the robotic is connected via Ethernet to two CPEs (HUAWEI B315s-22 and HUAWEI 5G CPE Pro Baloong 5000, respectively) that provide connectivity towards 4G and 5G networks. 5TONIC provides an implementation of 5G NSA (BB630 baseband and Advance Antenna System AIR 6488) and 4G (BBU5216 baseband and RRU 2203 with integrated antenna) provided by Ericsson. The benchmark of each RAT available at 5TONIC lab is provided in Table 4.

TABLE 4. WiFi, 4G and 5G NSA Benchmark at 5TONIC.

Metric	WiFi [2.4Ghz]	4G	5G NSA[7]
E2E Latency	1.77±0.68 ms	23.88±5.84 ms	6.56±1.04 ms
Packet-loss	0.11±0.35%	0%	0%
Data Rate (Uplink)	37.76±7.71 Mbps	44±0.20Mbps	96±1.81Mbps
Data Rate (Downlink)	49.39±1.80 Mbps	72±1.04 Mbps	600±13.50 Mbps
Jitter	1.10±0.73 ms	2.32±0.35 ms	0.46±0.18 ms

It is worth mentioning that, when the experiments were conducted, only 5G NSA was available as a 5G solution in 5TONIC. In the near-future, 5TONIC plans to deploy a 5G SA solution, as well as 5G mmWave radio. While SA solution will allow us to test the Digital Twin system under a native 5G core deployment, the mmWave radio will give us the opportunity to analyze the performance in the high-band spectrum.

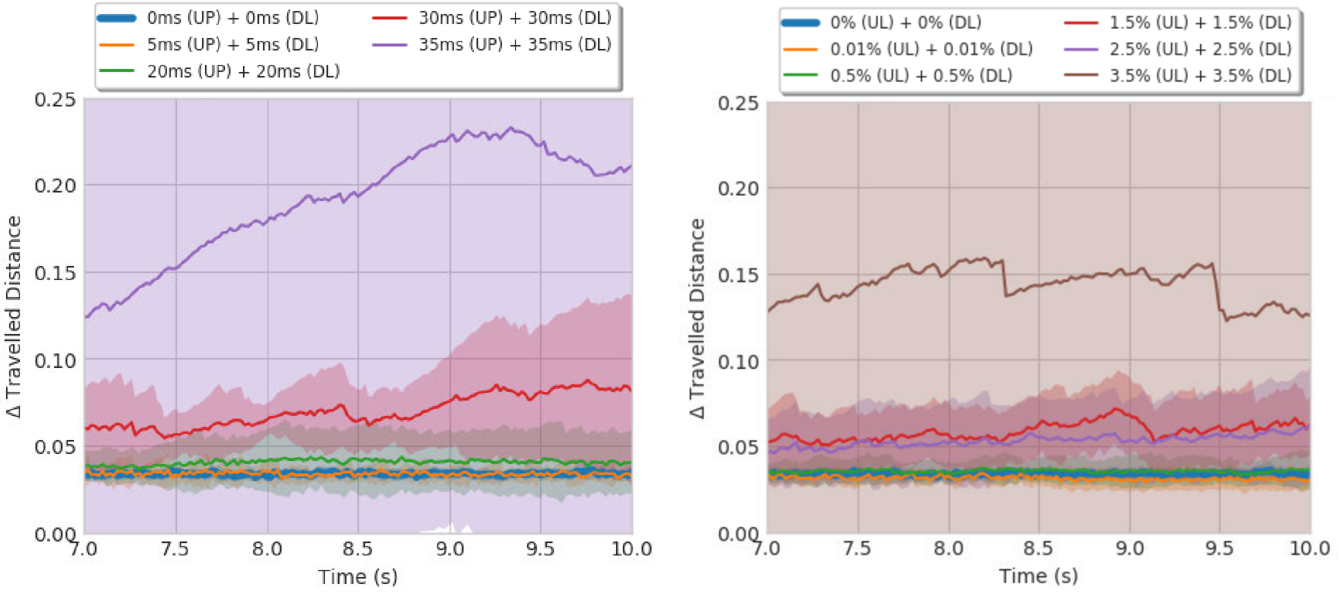
The first set of experiments assesses how the designed service operates over interference prone WiFi link, causing increased packet loss and jitter to occur. Throughout the duration of the experiment, the Digital Twin interacted with the *Control VNF* and 1-axis of the robotic arm is remotely controlled for period of 30 seconds with a control period of 20 ms and a movement offset of 0.010rad. The Digital Twin records the instructed and executed commands with which the travelled distance is computed. NetEm[8] is used to introduce symmetric artificial jitter and packet loss in uplink and downlink.

Figure 8 compares the average traveled distance for different values of jitter and packet loss on the WiFi link. Results show that a 5 ms jitter in uplink and downlink does not have any impact on the Digital Twin operation. This is because the 20 ms control and update period of the Digital Twin are not violated by such amount of jitter, allowing remote commands to be executed and the virtual replica to be updated

---

[7] 5G Massive MIMO Radio on 3.5GHz band; Downlink: 5G bandwidth 50MHz; Uplink Carrier Aggregation: 5G bandwidth 50MHz + LTE bandwidth 20MHz; Time Division Duplex (TDD) Pattern used is 7:3 (10:2:2).

[8] https://www.linux.org/docs/man8/tc-netem.html



(a) Jitter

(b) Packet Loss

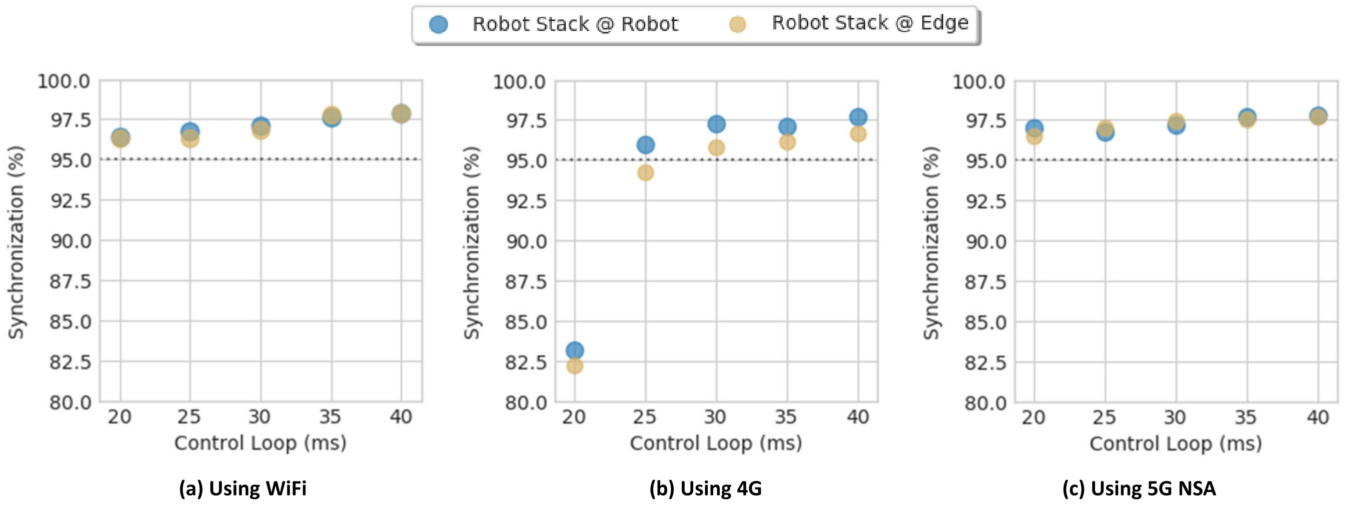
FIGURE 8. **Impact of network conditions on Digital Twin synchronization.**

in real time. With a 20 ms of uplink and downlink jitter, a slight increase of the deviation between the instructed and executed commands is noticeable, including the standard deviation of the measured values. This comes naturally since the control frequency constraint is not satisfied all times, causing some commands to be discarded by the robotic arm (since their lifespan is expired upon reception). Moreover, the joint state updates in the Digital Twin are delayed, resulting in a desynchronized virtual replica. As shown on Figure 8a, the deviation continues to increase with the jitter increase on the WiFi link. When the jitter is approximately double of the control period (e.g., 35 ms), the robotic arm starts to move erratically, including unpredictable jumping moves, which makes the whole environment unsafe for operation.

In turn, in Figure 8b, a similar analysis is provided by taking into account the packet loss. Although ROS uses TCP as the underlying transport protocol, a lost packet is never retransmitted within the 20 ms control period of the robotic arm (i.e., TCP_RTO_MIN is configured with 200 ms). With a packet loss of 0.01% in uplink and downlink, no visible impact is witnessed on the operation of the Digital Twin. The reasons are two-fold: *(i)* there is a very small number of packets lost; and *(ii)* it is able to recover when the lost packet is retransmitted. When the packet loss is increased to 0.5% in both directions, a slight deviation of the instructed and executed commands is witnessed. However, these values are still manageable by the Digital Twin, without a noticeable impact on its operation. However, this is not the case when the packet loss increases up to 1.5% and 3.5%. In the latter case, the operation of the robotic arm through the Digital Twin is degraded significantly, making it unusable.

The second set of experiments aimed to assess the minimum control period that can be supported in Digital Twin systems considering 4G and 5G generations of the cellular technologies, and WiFi. Throughout the duration of the experiments, the Digital Twin interacts with the *Control VNF* to remotely control 3-axis of the robotic arm for a period of 30 seconds with a movement offset of 0.010rad. In the Digital Twin, the synchronization between the robotic arm and its digital replica is measured over time.

Figure 9 compares the achieved synchronization by the Edge Robotic Digital Twin when implementing different control periods. Considering an average synchronization of 95% as the threshold below which the impact on the robotic arm operation is noticeable by a human-operator, it is not possible to operate the robotic arm with the minimum control period (i.e., 20 ms) using 4G. In fact, the operation of the robotic arm through its Digital Twin replica is only usable with a control period of ∼30 ms and ∼40 ms, when considering the whole ROS stack on the robotic arm or offloaded into the Edge, respectively. In turn, with 5G and interference free WiFi channel, the operation of the robotic arm is achieved using the minimum control period supported. The reason for such results is the lower latency witnessed in the 5G NSA system (∼6.56 ms) and WiFi (∼1.77 ms) when compared to the 4G system (∼23.88 ms). Still, for all experiments, whenever the distance increases (i.e., the latency between the Digital Twin and the physical device), the control period must be reconfigured to a higher value than the E2E latency. By increasing the control period, commands are issued at a lower frequency and, therefore, the robotic arm moves slower, allowing higher synchronization to be achieved over time. In addition, when the control and update periods are around or below the E2E



(a) Using WiFi (b) Using 4G (c) Using 5G NSA

FIGURE 9. Impact of control period variation.

latency, the control loop is continuously missing the desired rate and the *Control* and/or *Robot Drivers* VNFs may end up in error state. This error state will break the Digital Twin stack and the overall system will become unreliable and unstable.

VII. DISCUSSION

In this section, the results are briefly discussed and how they can be interpreted from the perspective of Industry 4.0. In parallel, research directions and open problems are identified.

A. DISTRIBUTED DIGITAL TWIN

The Edge Robotic Digital Twin incorporates Edge computing, NFV concepts and 5G connectivity in order to achieve real-time visualization and remote control. Its prototype implementation required the integration of several software components and virtualization technologies that actually enabled the functional split and distribution of the system. Computational offloading in the Edge is exploited with the experimental results providing insights with respect to magnitude of savings and performance improvements that can be achieved. The nature of the system can offer additional energy savings for the manufacturing sector by allowing dynamic resource allocation and service provisioning [47] for Operational Digital Twins.

B. TOWARDS WIRELESS FACTORY FLOOR

The implemented prototype also offers opportunities for computational offloading of time-sensitive component (e.g., *Control VNF*), not possible when leveraging on Cloud-based technologies. Edge-based technologies are a big aid to offload such time-sensitive components. However, in industrial environment where safety and reliability are of utmost importance, the behavior of the system could become unacceptable when facing increased latency and/or an unpredictable communication channel. As such, wireless technologies that support high reliable and low latency requirements is a current research direction to mitigate this issue [48], especially when moving the factory floor to be wireless by default. In addition, the initiative to bring Time-sensitive Networking (TSN) in industrial wireless [49] is also contributing towards the offloading of time sensitive components or applications.

C. 5G CONNECTIVITY FOR DIGITAL TWIN SYSTEMS

Real-time remote control with rapid visualization feedback is an example of such applications. In the implemented prototype, such behavior is truly achieved over 5G networks. Based on the experimental results, 5G enables faster and stable control and feedback frequencies that are not possible with other wireless technologies like 4G and interference prone WiFi, and directly contributes towards the realization of operational Digital Twins. However, the major deficit of the current 5G deployments appears to be the lack of support for industrial closed-loops that are around the lower latency bound (0.5 ms). 3GPP Release 16 [50] is expected to be the needed improvement for the 5G system in this direction. It will include URLLC enhancements for vertical services, being expected an increase adaptation of Digital Twins for industrial remote operation. Additionally, for high-bandwidth Digital Twins use cases that require high-definition video streaming or extended reality (i.e., VR and AR) utilizing the mmWave spectrum can bring great indoor coverage, even in a noisy and industrial environment as well as provide the high system capacity.

D. MULTI-RAT SUPPORT FOR INDUSTRIAL ENVIRONMENT

In a industrial environment, latency and transmission reliability can be improved by introducing multi-RAT connectivity. A physical device starts by selecting the RAT, among the available ones in the factory floor, that satisfies its connectivity requirements. Simultaneously, each physical device can also associate itself with another RAT to create backup communication links, thus improving the transmission reliability. However, the support of multiple flows in a transparent way for applications is a critical challenge when designing a Digital Twin system. One of the question that arises is at which layer of the protocol stack to implement such solution? Although recently standardized, Multipath TCP (MPTCP) [51] is a Transport layer protocol that can use one or more RAT interfaces to simultaneously achieve higher throughput and reliability. As such, MPTCP is gaining increasing popularity among vendors, telecom providers and startups which are finding applicability in Digital Twin solutions. Other solutions at lower layers, such as Packet Replication and Elimination Functions (PREF) [52] at the network layer or Frame Replication and Elimination for Reliability (FRER) [53] at Data Link Layer (only applicable to IEEE 802 link technologies), can also be leveraged to improve reliability.

E. CROSS-DISCIPLINARY TEAMING

The implementation of this Digital Twin prototype made us realize that experience from different engineering fields is needed in order to cope with the complexity of developing and integrating Digital Twin application. Tools and methods that are explicitly made for the fields of robotics, mechanical engineering, software and network development, system engineering and visual simulation are exploited in single system that needs to perform in an optimal way. This is a major obstacle in the applicability of Digital Twins in Industry 4.0. Research and software/hardware development work will need to become cross-disciplinary taking into consideration the needs from different fields.

VIII. CONCLUSION AND FUTURE WORK

In this work, an Edge Robotics Digital Twin, embodying the concept of Digital Twin as a Service, is implemented and evaluated in order to study the impact and the trade-offs of emerging Information and Communication Technologies. The implemented prototype relies on an open-source robotics middleware suite (ROS), making it flexible to easily accommodate different robotic systems. ROS is estimated to be supported by 55% of the world's robots by 2024, thus will play a key role on future robotic systems. First through a mathematical analysis and later verified by means of experimental validation in a Technology Readiness Level (TRL) 5-6, 5G and Edge computing are identified as must have technologies to support the most demanding Digital Twin use cases. Although WiFi is able to accommodate most of the requirements in ideal conditions, its susceptibility to interference make it not suitable for industrial and critical environments. In this sense, WiFi 6E appears as a potential candidate, but further experimental evaluations are still required. Notwithstanding, offloading robotic functions to the Edge introduced potential savings of 16% and 34% in terms of CPU and MEM usage, respectively. Lastly, further validation stages comprising not only real industrial machinery but also all the factory environments are required to validate the feasibility of yet to be commercial solutions.

Artificial intelligence (AI) and Machine Learning (ML) are being considered as short-term follow-ups of this work to provide the smart and automation capabilities envisioned by the Industry 4.0. The huge amount of data generated by industrial processes and exchanged through the Digital Twin solutions can then be crunched by AI/ML methods which outcomes will pave the way for novel industrial applications as well as optimizations on the resource utilization and energy consumption.

REFERENCES

[1] M. Hermann, T. Pentek, and B. Otto, "Design principles for industrie 4.0 scenarios," in *Proc. 49th Hawaii Int. Conf. Syst. Sci. (HICSS)*, Jan. 2016, pp. 3928–3937.
[2] A. A. F. Saldivar, Y. Li, W. Chen, Z. Zhan, J. Zhang, and L. Y. Chen, "Industry 4.0 with cyber-physical integration: A design and manufacture perspective," in *Proc. 21st Int. Conf. Autom. Comput. (ICAC)*, Sep. 2015, pp. 1–6.
[3] F. Tao and Q. Qi, "New IT driven service-oriented smart manufacturing: Framework and characteristics," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 1, pp. 81–91, Jan. 2019.
[4] F. Tao, L. Zhang, Y. Liu, Y. Cheng, L. Wang, and X. Xu, "Manufacturing service management in cloud manufacturing: Overview and future research directions," *J. Manuf. Sci. Eng.*, vol. 137, no. 4, pp. 1–11, Aug. 2015.
[5] D. Wu, M. J. Greer, D. W. Rosen, and D. Schaefer, "Cloud manufacturing: Strategic vision and state-of-the-art," *J. Manuf. Syst.*, vol. 32, no. 4, pp. 564–579, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0278612513000411
[6] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: The next computing revolution," in *Proc. Design Automat. Conf.*, Jun. 2010, pp. 731–736.
[7] B. R. Barricelli, E. Casiraghi, and D. Fogli, "A survey on digital twin: Definitions, characteristics, applications, and design implications," *IEEE Access*, vol. 7, pp. 167653–167671, 2019.
[8] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," *Int. J. Adv. Manuf. Technol.*, vol. 94, nos. 9–12, pp. 3563–3576, Feb. 2018.
[9] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.
[10] *Multi-Access Edge Computing (MEC); Framework and Reference Architecture*, document GS 003 v2.1.1, Group Specification, European Telecommunications Standards Institute (ETSI), 2019.
[11] *Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Architectural Framework Specification*, document GS 006 v2.1.1, Group Specification, European Telecommunications Standards Institute (ETSI), 2021.
[12] *Digital Enhanced Cordless Telecommunications (DECT); Study on URLLC use Cases of Vertical Industries for DECT Evolution and DECT-2020*, document TR 103 515 v1.1.1, European Telecommunications Standards Institute (ETSI), 2018.
[13] *Service Requirements for Cyber-Physical Control Applications in Vertical Domains*, document TS 22.104 v17.4.0, Technical Specification, 3rd Generation Partnership Project (3GPP), 2020.
[14] *5G for Connected Industries and Automation, Second Edition*, 5G Alliance Connected Industries Automat. (5G ACIA), Frankfurt, Germany, Feb. 2019.
[15] *5G E2E Technology to Support Verticals URLLC Requirements*, Next Generation Mobile Networks (NGMN), Final Delivrable v2.5.4, Frankfurt, Germany, 2020.
[16] A. M. Alba, J. H. G. Velásquez, and W. Kellerer, "An adaptive functional split in 5G networks," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 410–416.
[17] M. Grieves and J. Vickers, *Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems*. Cham, Switzerland: Springer, 2017, pp. 85–113.
[18] F. Tao and M. Zhang, "Digital twin shop-floor: A new shop-floor paradigm towards smart manufacturing," *IEEE Access*, vol. 5, pp. 20418–20427, 2017.

[19] F. Pires, A. Cachada, J. Barbosa, A. P. Moreira, and P. Leitão, "Digital twin in industry 4.0: Technologies, applications and challenges," in *Proc. IEEE 17th Int. Conf. Ind. Inform. (INDIN)*, vol. 1, Jul. 2019, pp. 721–726.

[20] *International Energy Outlook 2017*, US Energy Inf. Admin. (EIA), Report (R) IEO2017, Washington, DC, USA, 2017.

[21] W. Dai, H. Nishi, V. Vyatkin, V. Huang, Y. Shi, and X. Guan, "Industrial edge computing: Enabling embedded intelligence," *IEEE Ind. Electron. Mag.*, vol. 13, no. 4, pp. 48–56, Dec. 2019.

[22] A. Willig, K. Matheus, and A. Wolisz, "Wireless technology in industrial networks," *Proc. IEEE*, vol. 93, no. 6, pp. 1130–1151, Jun. 2005.

[23] J. P. Thomesse, "Fieldbus technology in industrial automation," *Proc. IEEE*, vol. 93, no. 6, pp. 1073–1101, Jun. 2005.

[24] J. D. Decotignie, "Ethernet-based real-time and industrial communications," *Proc. IEEE*, vol. 93, no. 6, pp. 1102–1117, Jun. 2005.

[25] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital twin in industry: State-of-the-art," *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019.

[26] A. Fuller, Z. Fan, C. Day, and C. Barlow, "Digital twin: Enabling technologies, challenges and open research," *IEEE Access*, vol. 8, pp. 108952–108971, 2020.

[27] S. Weyer, T. Meyer, M. Ohmer, D. Gorecky, and D. Zühlke, "Future modeling and simulation of CPS-based factories: An example from the automotive industry," *IFAC-PapersOnLine*, vol. 49, no. 31, pp. 97–102, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896316328397

[28] R. Rosen, G. Von Wichert, G. Lo, and K. D. Bettenhausen, "About the importance of autonomy and digital twins for the future of manufacturing," *IFAC-Papers OnLine*, vol. 48, no. 3, pp. 567–572, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896315003808

[29] J. Vachálek, L. Bartalsky, O. Rovný, D. Sismisova, M. Morhac, and M. Loksik, "The digital twin of an industrial production line within the industry 4.0 concept," in *Proc. 21st Int. Conf. Process Control (PC)*, Jun. 2017, pp. 258–262.

[30] S. Konstantinov, M. Ahmad, K. Ananthanarayan, and R. Harrison, "The cyber-physical e-machine manufacturing system: Virtual engineering for complete lifecycle support," *Procedia CIRP*, vol. 63, pp. 119–124, Jan. 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827117301324

[31] T. H.-J. Uhlemann, C. Lehmann, and R. Steinhilper, "The digital twin: Realizing the cyber-physical production system for industry 4.0," *Procedia CIRP*, vol. 61, pp. 335–340, Jan. 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827116313129

[32] R. Söderberg, K. Wärmefjord, J. S. Carlson, and L. Lindkvist, "Toward a digital twin for real-time geometry assurance in individualized production," *CIRP Ann.*, vol. 66, no. 1, pp. 137–140, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0007850617300380

[33] M. Schluse, M. Priggemeyer, L. Atorf, and J. Rossmann, "Experimentable digital twins—Streamlining simulation-based systems engineering for industry 4.0," *IEEE Trans. Ind. Informat.*, vol. 14, no. 4, pp. 1722–1731, Apr. 2018.

[34] O. Givehchi, H. Trsek, and J. Jasperneite, "Cloud computing for industrial automation systems—A comprehensive overview," in *Proc. IEEE 18th Conf. Emerg. Technol. Factory Automat. (ETFA)*, Sep. 2013, pp. 1–4.

[35] W. Xu, J. Cui, L. Li, B. Yao, S. Tian, and Z. Zhou, "Digital twin-based industrial cloud robotics: Framework, control approach and implementation," *J. Manuf. Syst.*, vol. 58, pp. 196–209, Jan. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0278612520301230

[36] P. D. U. Coronado, R. Lynn, W. Louhichi, M. Parto, E. Wescoat, and T. Kurfess, "Part data integration in the shop floor digital twin: Mobile and cloud technologies to enable a manufacturing execution system," *J. Manuf. Syst.*, vol. 48, pp. 25–33, Jul. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S027861251830013X

[37] L. Hu, N.-T. Nguyen, W. Tao, M. C. Leu, X. F. Liu, M. R. Shahriar, and S. M. N. Al Sunny, "Modeling of cloud-based digital twins for smart manufacturing with MT connect," *Procedia Manuf.*, vol. 26, pp. 1193–1203, Jan. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S235197891830831X

[38] C. Zhang, G. Zhou, J. He, Z. Li, and W. Cheng, "A data- and knowledge-driven framework for digital twin manufacturing cell," *Procedia CIRP*, vol. 83, pp. 345–350, Jan. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827119306985

[39] H. Huang, L. Yang, Y. Wang, X. Xu, and Y. Lu, "Digital twin-driven online anomaly detection for an automation system based on edge intelligence," *J. Manuf. Syst.*, vol. 59, pp. 138–150, Apr. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0278612521000467

[40] C. Liu, L. Le Roux, C. Körner, O. Tabaste, F. Lacan, and S. Bigot, "Digital twin-enabled collaborative data management for metal additive manufacturing systems," *J. Manuf. Syst.*, vol. 2020, pp. 1–33, May 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0278612520300741

[41] H. X. Nguyen, R. Trestian, D. To, and M. Tatipamula, "Digital twin for 5G and beyond," *IEEE Commun. Mag.*, vol. 59, no. 2, pp. 10–15, Feb. 2021.

[42] J. Cheng, W. Chen, F. Tao, and C.-L. Lin, "Industrial IoT in 5G environment towards smart manufacturing," *J. Ind. Inf. Integr.*, vol. 10, pp. 10–19, Jun. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2452414X18300049

[43] L. Girletti, M. Groshev, C. Guimarães, C. J. Bernardos, and A. de la Oliva, "An intelligent edge-based digital twin for robotics," in *Proc. IEEE Globecom Workshops (GC Wkshps*, Dec. 2020, pp. 1–6.

[44] K. Židek, J. Pitel', M. Adámek, P. Lazorík, and A. Hošovský, "Digital twin of experimental smart manufacturing assembly system for industry 4.0 concept," *Sustainability*, vol. 12, no. 9, p. 3658, May 2020. [Online]. Available: https://www.mdpi.com/2071-1050/12/9/3658

[45] H. Laaki, Y. Miche, and K. Tammi, "Prototyping a digital twin for real time remote control over mobile networks: Application of remote surgery," *IEEE Access*, vol. 7, pp. 20325–20336, 2019.

[46] J. Shuja, K. Bilal, S. A. Madani, M. Othman, R. Ranjan, P. Balaji, and S. U. Khan, "Survey of techniques and architectures for designing energy-efficient data centers," *IEEE Syst. J.*, vol. 10, no. 2, pp. 507–519, Jun. 2016.

[47] M. Afrin, J. Jin, A. Rahman, A. Rahman, J. Wan, and E. Hossain, "Resource allocation and service provisioning in multi-agent cloud robotics: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 2, pp. 842–870, 2nd Quart., 2021.

[48] P. Popovski, C. Stefanovic, J. J. Nielsen, E. de Carvalho, M. Angjelichinoski, K. F. Trillingsgaard, and A.-S. Bana, "Wireless access in ultra-reliable low-latency communication (URLLC)," *IEEE Trans. Commun.*, vol. 67, no. 8, pp. 5783–5801, Aug. 2019.

[49] S. Bush and G. Mantelet, "Industrial wireless time-sensitive networking: RFC on the path forward," Avnu Alliance, Beaverton, OR, USA, White Paper, Version 1.0.3, 2018.

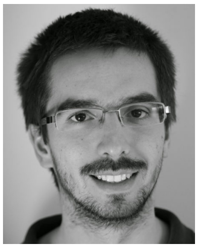
[50] *Study on Enhancement of 5G System (5GS) for Vertical and Local Area Network (LAN) Services (Release 16)*, document TS 23.734 v16.2.0, Technical Report, 3rd Generation Partnership Project (3GPP), 2019.

[51] S. Barré, C. Paasch, and O. Bonaventure, "Multipath TCP: From theory to practice," in *NETWORKING*, J. Domingo-Pascual, P. Manzoni, S. Palazzo, A. Pont, and C. Scoglio, Eds. Berlin, Germany: Springer, 2011, pp. 444–457.

[52] N. Finn, P. Thubert, B. Varga, and J. Farkas, "Deterministic networking architecture," IETF, Wilmington, DE, USA, Tech. Rep. RFC 8655, Oct. 2019. [Online]. Available: https://rfc-editor.org/rfc/rfc8655.txt

[53] N. Finn, "Introduction to time-sensitive networking," *IEEE Commun. Standards Mag.*, vol. 2, no. 2, pp. 22–28, Jun. 2018.

MILAN GROSHEV received the B.S. degree in telecommunication engineering from the Saints Cyril and Methodius University of Skopje, Macedonia, in 2008, and the M.S. degree in telecommunication engineering from the Politecnico di Torino, Turin, Italy, in 2016. He is currently pursuing the Ph.D. degree in telematics engineering with University Carlos III Madrid (UC3M), Spain. His Ph.D. research investigates the integration of edge and fog in virtual environments with objective to build lightweight, low-cost, and smarter robots.



CARLOS GUIMARÃES received the M.Sc. degree in computer and telematics engineering from the Universidade de Aveiro, Portugal, in 2011, and the Ph.D. degree in computer science, in 2019, under the scope of MAP-I Doctoral Program, Portugal. He is currently a Postdoctoral Researcher with Universidad Carlos III de Madrid (UC3M), where he is doing research activities in 5G technologies and main enablers, and the application of AI/ML into computer networks.

ANTONIO DE LA OLIVA (Member, IEEE) received the degree in telecommunications engineering, in 2004, and the Ph.D. degree from Universidad Carlos III Madrid (UC3M), Spain, in 2008.

Since 2008, he has been an Associate Professor with UC3M. He has published more than 30 articles on different networking areas. He is currently an Active Contributor to IEEE 802, where he served as the Vice-Chair of IEEE 802.21b and a Technical Editor of IEEE 802.21d. He served as a Guest Editor of *IEEE Communications Magazine*.

ROBERT (BOB) GAZDA (Member, IEEE) received the B.S. degree in electrical engineering from Drexel University, the M.S.E. degree from Carnegie Mellon University, and the M.S. degree in computer engineering from Villanova University. He is currently with InterDigital, where he is leading research and innovation focused on 5G and 6G network architectures and edge computing. He is also a Senior Director with the Technology Evolution and Prototyping Group, InterDigital's Wireless Networking Laboratory. He is also an accomplished engineering professional and a technologist with over 20 years of industry experience in wireless telecommunications, networking, and embedded systems. His research interests include wireless communications and networking, mobility, edge and distributed computing, and real-time systems.

• • •