DeSiRe-NG: An Architecture for Autonomous and Assisted 5G Network Measurement and Emulation

Simon Welzel*, Niels Hendrik Fliedner†, Maxim Friesen†, Christian Tismer‡, Philip Mildner‡, Syed Abdullah Rizvi‡, Claudius Noack¶, Lukas Dalhoff§, Henning Trsek† and Florian Klingler*
*Dept. of Computer Science and Automation, TU Ilmenau, Germany
†inIT - Institute Industrial IT, THOWL, Germany
‡Nuromedia GmbH, Germany § InnoZent OWL e.V., Germany ¶Lufthansa Industry Solutions, Germany

***Abstract*—5G (and beyond) cellular networks have gained much attraction in the past for various application domains in context of industry automation, autonomous and cooperative mobility, as well as smart cities. A main commonality of those use cases is their strong reliance on specific Quality-of-Service (QoS) metrics, which are crucial for seamless operation of those applications. This especially holds true for 5G use cases in the aviation domain, where a prominent application is the Virtual Table Inspection (VTI) – a process in which aircraft maintenance, disassembly, and reassembly are continuously monitored in real time via high-quality video and audio streams, enabling end-to-end tracking of quality control for all aircraft components. Providing concurrent high-quality video streams requiring high data rates and low latencies together with guaranteed packet delivery ratios is a main challenge of 5G networks – that particularly holds for 5G campus networks which are often difficult to scale. Our project *DeSiRe-NG* tackles exactly these shortcomings by providing a versatile architecture to detect under-performance of 5G networks. Further, our proposed architecture allows to generate the Digital Twin (DT) of a 5G network's QoS, which helps to improve the research and development process of novel networking architectures and applications, as well as aid the continuous supervision of a network. In a first proof of concept study, we present the functionality of our system in an industrial environment.**

I. Introduction

The increasing reliance on wireless networks across various industrial sectors requires reliable and high-performance wireless communication. This is especially relevant for critical application scenarios such as the Virtual Table Inspection (VTI) of aircraft engine parts [1]. In this specific use case, aircraft turbine inspections are initiated by Lufthansa Technik technicians, who live stream video footage of relevant disassembled engine parts to remote customers via the *AVIATAR* platform. These inspections require high-resolution and low-latency video streams that meet strict Key Performance Indicator (KPI) requirements. Wireless Local Area Networks (WLANs) are commonly used, but covering large areas requires multiple access points. Performing handovers between these can lead to temporal degradation of network performance [2], causing the loss of critical visual details. As a result, VTIs in unstable networks might require restarting, posing financial risks. Replacing WLAN with a 5G network reduces latency and increases data rates, but it remains essential to assess the capability of the 5G network to meet the Quality-of-Service (QoS) requirements for this use case. As more and more 5G networks are installed, the risk of interference generated by overlapping networks increases. To monitor and analyze QoS metrics, a measurement and monitoring system for detecting anomalies is needed, so mitigation measures can be employed before the live production environment is affected. This system needs to be capable of performing spatio-temporal measurements of relevant application metrics, as the location and time of the measurement device are relevant to the use case. Additionally, scalability is a mandatory requirement to enable simultaneous measurements across multiple points of interest.

In this paper, we address these challenges by proposing a system architecture for a scalable distributed measurement and monitoring system, capable of spatio-temporal end-to-end measurements. Based on the collected measurement and location data, algorithms determine when and where to perform new measurements to build a map of network performance and, subsequently, a Digital Twin (DT) of the QoS of the 5G network. The QoS measurement system is deployed on an Automated Guided Vehicle (AGV). User Equipments (UEs), such as 5G-enabled mobile phones, can detect initial QoS violations and trigger a Measurement Campaign, prompting the AGV to navigate the affected area and perform detailed QoS assessments. Based on the collected network QoS metrics, the system can emulate network characteristics using network impairment, facilitating reproducible application testing under controlled conditions. By intentionally inducing under-performance scenarios, such as introducing interference at the Physical Layer (PHY), we can assess the resilience of 5G networks against external disturbances. This is particularly important for applications such as the VTI, which rely on a stable connection.

This paper is structured as follows: First, we give an overview of related work in this research area. Then, we describe our system architecture, followed by the description of our prototypical implementation. Finally, we conclude the results of our paper and give an outlook into possible future work.

This work received funding from the German Federal Office for Information Security (BSI) and the German Aerospace Center (DLR) under grant agreement No. 01MO23028C.

IEEE International Conference on Factory Communication Systems (WFCS 2025)

II. Related Work

Our proposed system consists of multiple components. While prior research has explored these components separately, existing solutions remain fragmented, limiting their ability to be practically deployed in 5G environments. To the best of our knowledge, our work is the first to integrate real-time network performance measurements, radio map generation and DT-based impairment modeling into a single, scalable system. This section provides an overview of related work on individual components, highlighting key differences from our approach.

Network Performance Measurement Systems: Various approaches for network performance measurement systems exist in the literature. He et al. [3] propose a distributed measurement system, but it lacks a centralized entity, making scalability challenging. Our approach addresses this by combining distributed measurement with centralized orchestration, enabling scalable, real-time QoS monitoring and anomaly detection. Chai et al. [4] introduce a measurement system with central control, but without end-to-end measurements. QoS-intensive applications, such as the VTI, require full-path performance monitoring. Our system supports full end-to-end spatio-temporal QoS measurements. In our previous work [5], we introduced a system that supports both end-to-end measurements and scalability. We use an extended variant of this approach as a component for our system described in this paper.

Radio Map Generation and Interpolation: Extensive research has been conducted on generating *radio maps*, which visualize wireless channel measurements [6], [7], [8], [9]. These are widely used for localization in WLANs through Received Signal Strength Indication (RSSI) measurements. Since manually creating dense measurement maps is labor-intensive, research has also explored automation [8], [9] and interpolation techniques for sparse radio maps [7]. However, most studies focus on WLAN-based radio maps, while 5G radio map generation remains relatively underexplored. Additionally, existing research on 5G radio maps primarily appears in the context of indoor localization [10]. While existing works mainly rely on static measurement techniques, our approach introduces autonomous, spatio-temporal measurements to construct radio maps dynamically.

Network Testing: Three main approaches exist for testing applications in a network: Field tests, simulations and emulations. Field tests provide the most accurate results but are expensive, as they require deploying applications in real environments, potentially disrupting production processes. While network configurations can be adjusted, modifying infrastructure is costly, making field tests impractical for cost-effective and fast-paced development and testing of network applications.

Simulations model all network components' behaviors, allowing scalable and low-cost experimentation with different scenarios and parameters. However, simulations inherently lack accuracy due to their reliance on simplified models. A widely used 5G network simulator is the *OMNeT++*-based *Simu5G* [11].

Emulations integrate real hardware with simulated network components. For instance, in a VTI emulation, a physical camera and display could be connected to the emulator, while the emulator artificially degrades network conditions, e.g. by introducing packet loss, capping data rates or delaying packets. There exist various multi-purpose emulators for 5G systems, e.g. by Nardini et al. [12]. The emulator *EMULRADIO4RAIL*, proposed by Berbineau et al. [13], is designed for testing applications without on-site presence, focusing on measuring application performance rather than network characteristics. Network emulation is widely used in academia to analyze applications and network protocols regarding their behavior under varying network conditions. A commonly used tool is *NetEm*, which is integrated in the Linux kernel [14].

Both simulation and emulation rely on models that approximate real network behavior, limiting their accuracy. They cannot account for external influences, such as interference from other 5G networks, faulty components, or malicious attacks. We propose a system that bridges the gap between simulation and emulation by enabling data-driven emulation, where the impairment engine dynamically adjusts network conditions based on actual measured performance rather than abstract models.

Unlike existing works that focus on theoretical models or controlled lab environments, our approach is validated in an industrial setting through the VTI use case. This real-world deployment underscores the practical applicability of our proposed system architecture and demonstrates its ability to improve mission-critical 5G applications beyond simplified simulations.

III. System Architecture

In this paper, we propose the DeSiRe-NG system architecture, a distributed system for performing measurements of the QoS metrics accessible in a wireless network, analyzing the results and recreating QoS conditions in the form of network impairment. The QoS measurements are used to create a QoS DT of the system under test. The proposed measurement system is technology-agnostic, but limited to wireless communication technologies that don't interfere with the Out-of-Band (OOB) communication described below.

Several user categories interact with the DeSiRe-NG system: The *Operator*, who is present on-site, starts and stops measurement campaigns, and manually sends the AGV to locations using the manual operation mode. The *Technician* is responsible for conducting the VTI and runs the DeSiRe-NG *Mobile Application* on his UE. The *Network Administrator* optimizes the wireless network and resolves issues the DeSiRe-NG system detects. Finally, the *Researchers, Engineers and Developers* are usually not on-site, but test their applications using the Impairment Entity.

The measurements are conducted using a distributed measurement system, the Measurement Entity. To obtain measurements from a whole area and not just statically from a single position, we use a physical device that performs the actual measurements —the Measurement Client (MC) —that is mounted on an AGV, which is roaming the facility. The AGV is controlled either

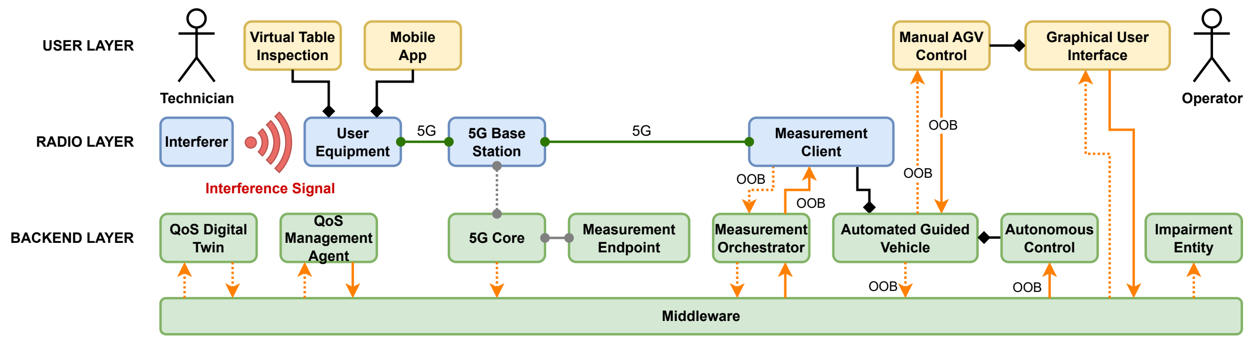


Figure 1. Proposed architecture and data flow (colored arrows). Orange dotted arrow: Measurement and location data; Orange solid arrow: Commands and Control; Gray dots: 5G Backbone; Green dots: 5G radio interface; Black diamond: Composites. OOB: Out-of-band communication.

autonomously or manually by an operator through the Graphical User Interface (GUI).

The measurement data is transferred to the QoS DT, which stores all measurement data centrally. The GUI visualizes the measurement data from the QoS DT and allows both real-time monitoring as well as inspecting previous measurement data.

Researchers or developers testing an application off-site use the GUI to select a location on a map of the facility where the measurements were conducted. The Impairment Entity manipulates the connection between two freely selectable devices to test any application without on-site presence.

An overview of the proposed system architecture is given in Figure 1. In the following, the architecture's components are described in more detail.

A. *Quality of Service Digital Twin*

A DT serves as a virtual model of a real-world entity. In a smart factory facility, a DT of the complete factory would contain, for example, the physical layout, the network, the application QoS, and possibly many additional components. Our project provides one of these components, a DT of the 5G Radio Access Network (RAN) QoS. This DT is created using the QoS metrics measured by the Measurement Entity and can be utilized in the following three ways: First, the user can visualize the spatio-temporal QoS metrics of the facility on a heat map. Second, the QoS Management Agent monitors the QoS DT for QoS violations, notifies other system components accordingly, and decides whether further measurements might be necessary. Third, the data in the QoS DT can be used with the Impairment Entity to impair the connection between devices on the network level, for example to evaluate a system's resilience against previously recorded QoS degradation patterns.

The QoS DT allows to query data for any desired location. If there are no Measurement Points stored for a queried location, the values are interpolated, which requires the existence of Measurement Points in the vicinity. Otherwise, we perform extrapolation based on channel models and physical space information, if available. This will also allow the selection of locations outside the range of existing Measurement Points.

B. *Measurement Entity*

The *Measurement Entity* is a distributed sub-system used for conducting the measurements. For each facility a separate Measurement Entity is used. It consists of three components: At least one MC—a mobile device that is moved inside the area covered by the wireless network under test. Every MC measures the network performance by connecting to the second separate entity, the Measurement Endpoint (ME), via the wireless network under test. The ME serves as endpoint for the network performance measurements. While the MC needs to be a physical device, as it is supposed to move around and access the wireless network from different places, the ME only needs to be connected to the tested network, allowing for virtualization. Using multiple MC-ME pairs is supported.

The measurements are coordinated by the third component, the Measurement Orchestrator (MO), which sends instructions to all local MCs when starting or stopping the measurements. Whenever new measurement data is recorded on an MC, it is sent to the MO via an OOB connection. If this data was routed through the system under test, it would compromise the measurement results. Thus, a second wireless network connection that does not interfere with the wireless network under test is required. While both the MC and the MO require such a connection, the ME does not, as the control instructions are transmitted before and after the actual measurements are conducted, but never during the recording of measurement data. We use a centralized MO on a dedicated device for two reasons: First, it allows using multiple MCs at the same time for certain use cases. Second, the MO is required to always provide connectivity to the MCs, as the OOB network might not always have an uplink to the internet. However, the MO can be set up in a way that it has access to both the internet and the OOB network. In case of internet connectivity issues, the MO is able to buffer measurement data until connectivity is restored. This enables temporary standalone operation, increasing the flexibility. The two measured performance metrics are data rate and latency.

C. *Middleware*

The Middleware serves as a conveyor layer between various components of the DeSiRe-NG system (see Figure 1). It stores an overview over existing Measurement Campaigns and the network addresses of known MOs. Incoming commands from the GUI to start and stop measurements are forwarded to the corresponding facility's MO. The MO in turn sends measurement data, which the Middleware matches to separately collected positioning data, attributes the correct Measurement Campaign and finally stores the now complete Measurement Point in the QoS DT. Measurement Points that belong together, usually because they have been recorded in succession, are grouped in a *Measurement Campaign*. This allows for later segregation and separate analysis to compare measurements taken at different times or with different parameters. A cloud-based Middleware is recommended, as it needs to be available to the other components at all times.

D. Automated Guided Vehicle

An AGV is a mobile robot that uses sensors, cameras, and software to navigate in an environment. In our architecture, the AGV collects positioning information, adds a timestamp to it and sends it to the Middleware, where the location data is matched to the measurement data. The AGV carries the MC, enabling it to collect measurement data in different locations. The AGV has three main states, which are also described in Figure 2:

1) Idle: The AGV is online but no Measurement Campaign is active. In this state, the AGV listens for new commands but does not perform any actions.
2) Autonomous data collection: The AGV autonomously navigates to a position after receiving the command from the QoS Management Agent, and the MC then collects the measurement data for the surrounding area.
3) Assisted data collection: The Manual AGV Control deploys the AGV to a specific coordinate by selecting a location on the map within the GUI. This mode overrides the autonomous data collection and the AGV will return to the autonomous mode after it has reached the requested position.

The autonomous control enables the usage of the AGV without operator interaction. It is activated when the QoS Management Agent detects a QoS degradation. Upon activation, the AGV is autonomously deployed, a Measurement Campaign is created and started, and the results are analyzed, with the aim of proposing a solution to the issue.

E. Graphical User Interface

The GUI allows the operator to start and stop Measurement Campaigns, inspect the live measurement data, view the measured performance metrics on a map or inspect single Measurement Points. The GUI is cloud-based to ensure constant availability. The QoS is visualized on a GUI with controls for the time range to be shown. The measurement data can be queried from the QoS DT via the Middleware and visualized as a heat map. Single Measurement Points as well as any location within the facility can be chosen to be viewed individually or to be used with the Impairment Entity. The GUI features the manual AGV control, allowing the Operator to direct the AGV to a position.

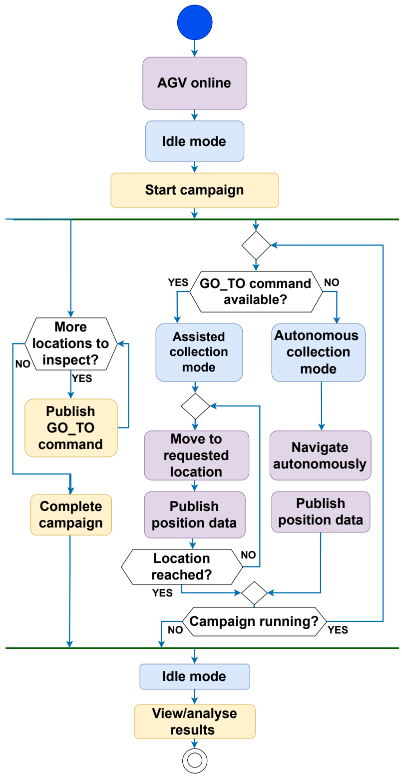


Figure 2. AGV command flow. Yellow: Operator action; Purple: AGV action; Blue: AGV mode

F. Mobile Application

A mobile application installed on each technician's UE serves as a tool for monitoring and visualizing the QoS. It supports two modes of measurements, an active mode and a passive mode. In active mode, the user can conduct latency, throughput, and jitter tests using the mobile application. In passive mode, the mobile application runs in the background and reports Reference Signal Received Power (RSRP), Reference Signal Received Quality (RSRQ), and Signal-to-Noise-plus-Interference Ratio (SINR) observed by the UE. If one of these values violates a predefined threshold, the user is notified.

If the mobile application detects a QoS degradation at any point, it notifies the technician and forwards the measurement results and its position to the QoS DT. In case of a QoS violation, the QoS Management Agent deploys the AGV to perform a Measurement Campaign at the corresponding position and update the QoS information within the DT. This enables administrators to react and mitigate the insufficient QoS situation. For example, the technician could move the VTI to a space with better QoS conditions. Additionally, comparing the past and present QoS situation provides future anomaly detection tools with the means to determine possible causes for the QoS degradation.

G. Impairment Entity

The purpose of the *Impairment Entity* is to allow the QoS DT to act upon other network devices. It emulates the spatio-temporal QoS behavior of the wireless network under test for later application testing. This enables testing applications off-site without the need to actually deploy them in the system under test, which accelerates system development and integration. The Impairment Entity connects two Ethernet devices and manipulates the exchanged packets according to location- and time-variant QoS data from the QoS DT for a specified Measurement Campaign. Furthermore, it displays the currently used parameters and allows to change them based on a graphical representation of the measured area. Thus, it is possible to select any location from the wireless network under test that the initial measurements have been conducted in. The Impairment Entity facilitates comprehensive network fault analysis through the emulation of real-world events, encompassing interference patterns, device malfunctions, and jamming attacks.

This functionality enables both post-mortem analysis of network faults and assessment of application resilience under realistic QoS conditions, all while eliminating the need for repeated physical interference and minimizing disruption to operational networks. Additionally, the Impairment Entity allows for simultaneous real-time testing of applications during an active measurement campaign. A detailed technical overview and performance analysis of the Impairment Entity is given in our recent work in [15].

H. PHY Level Impairment

The PHY level impairment simulates a jamming attack on the 5G system, which consequently affects the network's QoS by interfering with the data transmission. Additionally, selectively impairing the 5G communication on the PHY level enables us to evaluate the QoS DT implementation. This component is optional and only used for evaluation or creating targeted PHY level impairment, when inducing radio channel fading characteristic changes is not feasible.

To avoid excessive power consumption and achieve a higher transmission amplitude instead, we chose a broad-band interference signal based on Orthogonal Frequency-Division Multiplexing (OFDM). The OFDM parameters are selected to either broadly disrupt the transmission of uplink and downlink user data (Physical Uplink Shared Channel (PUSCH) and Physical Downlink Shared Channel (PDSCH)), respectively, or more narrowly block Synchronization Signal Block (SSB) reception, in a brute-force attempt matching the regular jammer type defined by Arjoune and Faruque [16].

IV. Prototypical Implementation

This section describes the prototypical implementation of the system architecture presented in the previous section. A graphical overview is given in Figure 3. We published our source code for the prototypical implementation on GitHub [17].

A. Measurement Entity

The MC, MO and Middleware are set up on dedicated Raspberry Pi Model 4 devices. In order to connect to the 5G campus network, the MC mounted on the AGV uses a Nokia XR20 5G-capable smartphone in USB tethering mode. Since there is not yet positioning support for 5G, we collect position data separately through the AGV. The connection between MC and MO is realized with an OOB IEEE 802.11 WLAN.

For conducting the actual measurements, we use *iperf3* [18] and *ping* between MC and ME, with an interval of one second. The ME is virtualized in a Debian-based Virtual Machine (VM) on a device directly connected to the 5G campus network.

B. Automated Guided Vehicle

The AGV uses the Robot Operation System 2 (ROS2) Humble framework, which can be interacted with by external services. The model of the AGV is Robowork's Mecabot Pro, which is a four-wheeled AGV using Jetson Orin NX and an STM32-based microcontroller. It features a Leishen C16 16-line 3D Light Detection and Ranging (LiDAR) and an Orbbec Astra RGB depth camera.

Before the start of a Measurement Campaign, a grid map of the facility under test is generated and subsequently loaded into the AGV. This allows the AGV to orientate itself autonomously within the learned environment and navigate to requested coordinates. AGV commands and positioning data are communicated via the OOB interface. An overview of the flow of AGV commands is shown in Figure 2. Commands to start or stop data collection or to navigate to a given position are scheduled within the Autonomous Control and sent to the AGV sequentially. All commands must be explicitly acknowledged and completed by the AGV before the next command is scheduled. In error cases the AGV can also reject the execution of commands, e.g. if it is not able to navigate to a requested position.

C. Quality of Service Digital Twin

The QoS DT stores data in a MySQL-based MariaDB database. It interfaces to the other components via Representational State Transfer (REST) Application Programming Interfaces (APIs) to receive incoming new measurement data, as well as handing out previously stored measurement data. In our implementation, the QoS DT is installed on the same device as the Middleware.

D. Graphical User Interface

The GUI receives spatio-temporal QoS measurement data via the Middleware and consists of three web pages. These can be hosted on a server on premise or off-site in a cloud.

The primary web page displays a 2D view with a heat map of the monitored 5G network environment, overlayed on a floor plan of the facility. A React-based NextJS stack using Three.js and D3.js libraries is used for this visualization purpose. AGV positions and the corresponding measurement data are color-coded accordingly to visualize anomalies within the QoS of the 5G network. A time slider allows to explore

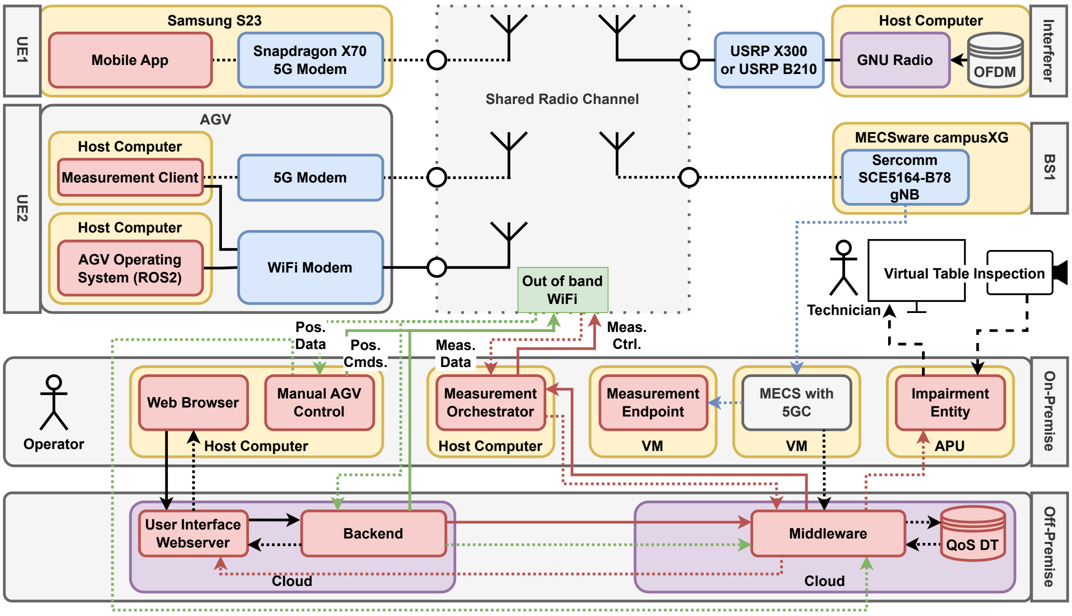


Figure 3. Implementation components. Yellow: Processing units; Red: DeSiRe-NG-specific components; Blue: Radio components; Dotted: Measurement Data; Solid: Command and Control; Dashed: Use Case Data

recorded measurement data over selected time periods. The Operator can interact with the heat map to view details such as measured data and campaign information. Measurement Campaigns can be managed through the second web page, which allows to start and stop a campaign, select an AGV and view the results. The third web page shows statistical charts of historical measurement data.

E. 5G System

The 5G network environment employed to integrate and validate the developed architecture is based on the MECSware *campusXG* kit. It contains a Sercomm SCE5164-B78 gNodeB (gNB), an indoor cell implementing both the 5G radio and the base band unit, operating at 3700–3800 MHz. The attached 5G Core Network (5GC) is based on the Polaris Networks *Unicorn* core and is deployed within the MECSware virtual machine cluster located on premise within an edge cloud environment.

F. Impairment Entity

The Impairment Entity is set up on an Accelerated Processing Unit (APU). One of the Ethernet ports serves as uplink and control port to receive control instructions, while two other ports connect the devices that are supposed to have their connection impaired. The connection between these devices is manipulated using NetEm [14]. For details on the Impairment Entity, please refer to our work in [15].

V. Proof Of Concept

With this proof of concept we demonstrate the interaction of our architecture components, based on the prototypical implementation described above. We tested the mapping of QoS measurement data to AGV locations, impairing a network with previously recorded QoS measurement data using the Impairment Entity, as well as interpolating and visualizing the QoS measurement data within our DT.

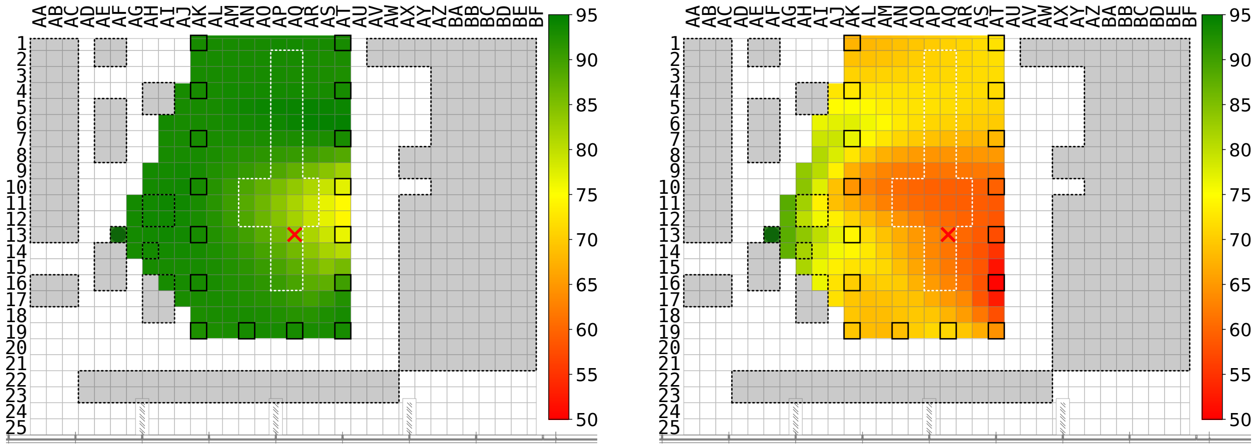
We conducted the integration tests at the *SmartFactoryOWL* in Lemgo, Germany. Real-time location data was provided by an AGV and transmitted to the MO.

For this purpose, a Roboworks's Mecabot Pro AGV was deployed, running ROS2 and the Adaptive Monte-Carlo Localizer in conjunction with the installed Leishen C16 LiDAR. The location data was temporally mapped to the network measurement samples, which were captured during our first Measurement Campaign runs within the on-site 5G campus network in the SmartFactoryOWL. In addition, the communication between GUI and Middleware, using the Middleware's API, has been tested successfully.

We used data measured in our 5G campus network to parameterize the impairment functionality of our system, conducted measurements by using the Impairment Entity for impairing the connection between MC and ME, and finally compared the original measurement data to the impaired measurement data, showing high fidelity. For details, please refer to our work in [15].

Lastly, we conducted a PHY-level impairment evaluation of the QoS DT in the SmartFactoryOWL. The setup for determining the QoS consisted of the 5G system described in Section IV-E and a Sierra Wireless *AirLink XR90* 5G router mounted to a Clearpath *Jackal* AGV, which served as a stand-in. For the 5G connection, we chose a default setting of 3.75 GHz center frequency, 100 MHz bandwidth and a 4:1 ratio between downlink and uplink resources.

For impairment, we used an Ettus Research *USRP B210* Software Defined Radio (SDR) connected to a Linux host computer running GNU Radio to emit a wide-band OFDM



(a) Heat map without jamming. (b) Heat map with jamming.

Figure 4. Interpolated QoS measurements in the SmartFactoryOWL, given in Mbit/s. Green circle: gNB position; Red X: Jammer position; Dotted outlines: Obstacles of at least the height of the gNB position.

jamming signal. To avoid a full Denial of Service (DoS), the jammer's center frequency and bandwidth was chosen to only cover part of the data-carrying channels, without impairing the SSB. We used the following OFDM parameters: Center frequency 3.775 GHz; subcarrier spacing 40 kHz; Fast Fourier Transform (FFT) length 512; transmission bandwidth 20 MHz; transmission power 80 % relative gain, which translates to a transmission power of 13 dBm at 3.775 GHz for the *USRP B210*.

The placement of the gNB, the jammer, and the obstacles present are visualized as part of the heat map in Figure 4. The raster are floor tiles with size 60 cm by 60 cm. Figure 5 shows data rate and RSSI for the individual Measurement Points without interpolation. To estimate the QoS state between Measurement Points we used cubic interpolation.

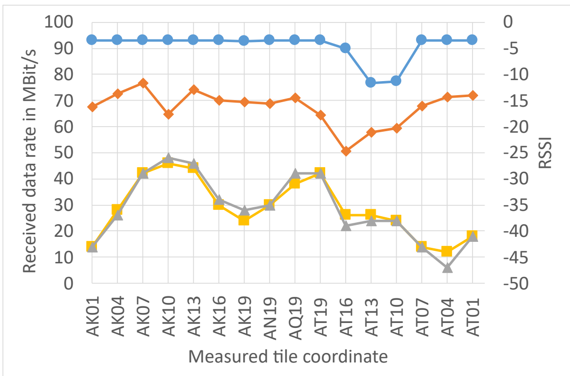


Figure 5. Results of QoS measurements in SmartFactoryOWL. Blue circle: Received data rate without jamming; Orange diamond: Received data rate with jamming; Grey triangle: RSSI without jamming; Yellow square: RSSI with jamming.

The QoS was determined using repeated iperf3 [18] User Datagram Protocol (UDP) network performance measurements between the *AirLink XR90* 5G router and an iperf3 server connected to the 5GC. For this measurement, the bandwidth parameter of iperf3 was set to 100 Mbit/s. This approach is consistent with our measurement system, as described in Section IV-A.

For each measurement, the AGV navigated to a set of equally distanced positions on the SmartFactoryOWL floor. At each position, we created two impairment situations: The first situation had no active impairment, but passive impairment from radio channel fading due to multipath propagation and shading. The second situation still had passive impairment, but we additionally caused active impairment using the OFDM jammer.

We interpreted the data rate as our main QoS indicator. The RSSI determined by the gNB served as our control to rule out transmission power fluctuation or radio channel attenuation influencing the data rate between measurements with and without active impairment.

As a result, without active impairment the data rate varied only slightly between the positions of the AGV due to radio channel fading, as seen in Figure 5. With active impairment, the data rate dropped significantly. Both situations are visualized as a heat map in Figure 4, where they are mapped to an excerpt of the SmartFactoryOWL we used for QoS measurements. While the data rate varied significantly between measurements with and without active interference, the RSSI remained mostly stable, which can be seen in Figure 5. Consequently, we could determine that the QoS variations were indeed caused by the active interference, instead of changes in transmission power or radio channel attenuation.

These results outline the ability of our QoS DT to represent the spatio-temporal QoS state of the facility's 5G network. Considering the RSSI being similar with and without activated jamming we can distinguish between passive and active impairment causes, namely radio channel fading of a known facility space and active interference, as demonstrated with a jammer. The interpolation allows to estimate the QoS state between Measurement Points.

Locating an active interferer such as a jammer is possible within limitations. Comparing the heat map of the unimpaired communication in Figure 4a sets expectations for QoS degradation due to radio channel fading and shadowing. The increased severity of the QoS degradation in Figure 4b would allow to estimate the rough location of the jammer, given that the physical radio channel has not been altered. However, due to only a single Measurement Campaign path being performed near the jammer, a trilateration-based localization becomes unlikely. More exact localization of the jammer requires additional measurements, which confirms the need to update the QoS information, as we stipulate in Section III-F.

Conversely, significant RSSI deviations in a repeated QoS Measurement Campaign would hint towards physical changes within the facility environment, influencing the radio channel propagation and fading characteristics. Relating to the VTI, a possible cause would be changing radio wave scattering patterns due to large metal objects from turbines being located or moved.

Known limitations of our implementation of the interpolation relate to the absence of a 3D model of the facility with known radio propagation characteristics. Such a model would enable us to more accurately estimate the radio channel fading between Measurement Points, for example by using ray tracing. A possible source of 3D models could be developed by integrating our QoS DT into an overarching smart factory DT, which already contains these models.

VI. Conclusion

In this paper, we gave an overview of our distributed, scalable QoS measurement system, enabling spatio-temporal end-to-end measurements in 5G and other networks. This measurement system assists in detecting QoS violations for use cases such as the VTI. We created a QoS DT with the measurement data, which allows for data exploration via a GUI and application testing via our Impairment Entity. In a proof of concept experiment, we showcased the opportunities our approach will create, as well as current limitations.

VII. Future Work

We plan to perform more in-depth analyses of our system's performance regarding the portrayed use case. Our QoS DT enables future tools to perform anomaly pattern analysis, for example to detect malicious interference, misconfigured systems or unintended environmental changes, all of which could affect the QoS during a VTI. For the Impairment Entity, the extrapolation functionality needs to be implemented and thoroughly evaluated by comparing it to actual measurement data, ensuring that the used models estimating path loss and its influence on network performance are accurate. Furthermore, we will compare various interpolation mechanisms and their fidelity.

The VTI is also subject to future enhancements, such as performing three-dimensional digitization of aircraft parts, creating a DT which can then be used for the inspection. This DT could feed the extrapolation functionality by computing path loss profiles and radio channel fading using ray tracing. As the turbine might cause a large part of the radio channel fading behavior and the camera with its 5G connection will always be close to it, our QoS DT's extrapolation ability would benefit from these enhancements.

References

[1] M. Patzold, "5G Unlocks Its Power for Global Mobile Connectivity [Mobile Radio]," *IEEE Vehicular Technology Magazine*, vol. 15, no. 2, pp. 5–11, 2020.

[2] L. Dimopoulou, G. Leoleis, and I. Venieris, "Fast handover support in a WLAN environment: challenges and perspectives," *IEEE network*, vol. 19, no. 3, pp. 14–20, 2005.

[3] Y. He, F. Wang, Q. Sun, L. Tian, and Y. Wang, "Research and design of distributed testing framework for 5G network," in *Proceedings of the 2023 4th International Conference on Computing, Networks and Internet of Things*, ser. CNIOT '23, Xiamen, China: Association for Computing Machinery, 2023, pp. 554–561.

[4] X. Chai, G. Cai, and H. Li, "aSIT: An Interface-oriented Distributed Automated Test System," in *Proceedings of the 2021 2nd International Conference on Computing, Networks and Internet of Things*, ser. CNIOT '21, Beijing, China: Association for Computing Machinery, 2021.

[5] N. Mondal, M. Niggemeyer, S. Welzel, and F. Klingler, "Poster: Decentralized Framework for Measuring End-to-End Performance in 5G Networks," in *2024 IEEE Vehicular Networking Conference (VNC)*, IEEE, 2024, pp. 265–266.

[6] S. H. Jung, B.-C. Moon, and D. Han, "Performance evaluation of radio map construction methods for Wi-Fi positioning systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 4, pp. 880–889, 2016.

[7] J. Bi et al., "Fast radio map construction by using adaptive path loss model interpolation in large-scale building," *Sensors*, vol. 19, no. 3, p. 712, 2019.

[8] W. Liu and J. Chen, "UAV-aided radio map construction exploiting environment semantics," *IEEE Transactions on Wireless Communications*, vol. 22, no. 9, pp. 6341–6355, 2023.

[9] S.-H. Jung and D. Han, "Automated construction and maintenance of Wi-Fi radio maps for crowdsourcing-based indoor positioning systems," *IEEE Access*, vol. 6, pp. 1764–1777, 2017.

[10] S. Huang, K. Zhao, Z. Zheng, W. Ji, T. Li, and X. Liao, "An Optimized Fingerprinting-Based Indoor Positioning with Kalman Filter and Universal Kriging for 5G Internet of Things," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, p. 9 936 706, 2021.

[11] G. Nardini, D. Sabella, G. Stea, P. Thakkar, and A. Virdis, "Simu5G – An OMNeT++ Library for End-to-End Performance Evaluation of 5G Networks," *IEEE Access*, vol. 8, pp. 181 176–181 191, 2020.

[12] G. Nardini, G. Stea, and A. Virdis, "Scalable real-time emulation of 5G networks with Simu5G," *IEEE Access*, vol. 9, pp. 148 504–148 520, 2021.

[13] M. Berbineau et al., "Zero on site testing of railway wireless systems: the Emulradio4Rail platforms," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, IEEE, Virtual Conference, 2021, pp. 1–5.

[14] S. Hemminger et al., "Network emulation with NetEm," in *Linux conf au*, vol. 5, 2005, p. 2005.

[15] N. Mondal, M. Niggemeyer, S. Welzel, and F. Klingler, "DecMEN: Scalable Measurement and Impairment Framework for Network Characterization in 5G (+)," in *2024 IEEE International Performance, Computing, and Communications Conference (IPCCC)*, IEEE, 2024, pp. 1–6.

[16] Y. Arjoune and S. Faruque, "Smart Jamming Attacks in 5G New Radio: A Review," in *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*, S. Chakrabarti and R. Paul, Eds., Nevada, US: IEEE, 2020, pp. 1010–1015.

[17] D.-N. Consortium. "GitHub: The DeSiRe-NG Consortium. "[Online]. Available: https://github.com/desire-ng-consortium.

[18] iperf3. "The TCP, UDP and SCTP network bandwidth measurement tool. "[Online]. Available: https://iperf.fr/.