DecMEN: Scalable Measurement and Impairment Framework for Network Characterization in 5G(+)

Niladri Mondal*, Mario Niggemeyer*, Simon Welzel* and Florian Klingler*
*Dept. of Computer Science, TU Ilmenau, Germany
{mondal, niggemeyer, welzel, klingler}@wnc-labs.org

***Abstract*—The fifth generation (5G) is the current state-of-the-art mobile communication system that has features for ensuring reliability and throughput being key ingredients for many application domains. A major distinction of applications using cellular networks is that they employ different requirements regarding the QoS, e.g., latency and data rates for communication. When adapting network protocol to the underlying communication system for increasing the reliability of applications, reproducible tests and measurements of network characteristics are of utmost importance to streamline a systematic development process of those applications. In this paper, we present a framework called Decentralized Measurement and Emulation of Networks (*DecMEN*) which is a versatile, decentralized and scalable tool for reproducible measurement and emulation of network characteristics of 5G and beyond networks. Along presenting a novel architecture for scalable measurements and network emulation, we perform a feasibility study on a private 5G campus network to showcase our developed framework in a real world setting. To assess the performance, we empirically evaluate the results of measurements of our 5G network and the emulation of network characteristics to observe differences of important networking metrics consisting of communication latency, and data rate. Our study reveals promising results, where the evaluation shows a good match of network characteristics between the real 5G network under test and the emulated 5G network.**

I. Introduction

The developments in recent cellular wireless communication technologies led to their implementation across diverse application domains, such as industrial automation, video streaming, vehicular communication, mission critical applications, and many more. This requires different Quality-of-Service (QoS) profiles of a given network in order to meet the requirements of such application domains. QoS metrics like throughput and latency play a crucial role to operate a network system in a given application domain. 5G is the state-of-the-art cellular wireless technology that was designed and standardized by Release 15 of 3GPP [1], which addresses exactly those challenges to offer flexible QoS profiles for resilient network operation. Beside public 5G networks, which consist of several base stations connected to a core network, nowadays also private 5G Campus networks gain high attraction. Particularly, the main motivation for such private 5G campus networks is that they can employ the rich feature-set of ultra reliable and low latency communication (URLLC) systems offered by 5G in a smaller – and private – context. This makes it a prime alternative to other wireless communication systems like IEEE 802.11 WLAN. To assess the impact of network characteristics on application performance, a thorough measurement and performance evaluation at the different distributed network elements is necessary in combination with reproducible evaluation of those applications.

Based on their nature, wireless communication systems can face different challenges of interference, channel loss, fading and shadowing, which could have a huge negative impact on the QoS level of the network. To study and analyze such wireless network channel anomalies in detail, reproducing the network characteristics is required. In particular, to obtain a reproducible representation of 5G network characteristics in a real world setting, two main methodologies are possible. A first approach would be to generate a detailed digital twin of the wireless network incorporating all effects from a wireless channel perspective. This digital twin of the environment from the perspective of the wireless channel characteristics can be used to emulate the behavior of the 5G network which then allows to have a reproducible representation of the network characteristics under a controlled environment (e.g., [2]). The major drawback of such an approach is that it employs detailed measurements of the wireless channel which often are complex and time consuming. Further, there is the risk that the emulation and simulation models of 5G networks could differ from 5G network elements used in real world scenarios, making the digital twin representation inaccurate. A second approach for creating a digital and reproducible representation of the 5G network characteristics is to purely focus on networking characteristics which are important for applications, e.g., latency, data rate, and packet delivery ratio. Thus, instead of modelling the wireless channel as accurate as possible which often is impossible to achieve at larger scale, application-critical network metrics are measured. Although at first sight such a methodology seems to be more inaccurate in comparison to reproducing detailed wireless channel conditions, this application-centric representation has many advantages. First, any possible network configuration can be used to create a digital representation of the networking characteristics, which includes even black-box testing where the internal behavior of a 5G network is unknown or not being revealed by a vendor. Second, such an approach not only incorporates effects on the network caused by channel conditions, but also incorporates non-negligible effects caused by other network users, e.g., interference and queuing delays of packets in the network. Once such a digital representation of the 5G network is available, a network emulation can reproduce the performance of the 5G network under test within a controlled and reproducible

43rd IEEE International Performance Computing and Communications Conference (IPCCC 2024)

environment. This allows an application to observe the same networking conditions in repeated experiments, which further allows developers to adapt networking protocols to deal with this networking conditions for increasing the resilience of that application.

In summary, the combination of application-driven measurement of network characteristics plays a significant role in the development process of applications to create resilient networking protocols for real world wireless networks under varying channel conditions to assess the application performance in a systematic, reproducible and representative way.

II. Related Work

The works from Ullah et al. [3] and Erunkulu et al. [4] discuss and identify the possible use cases of 5G in context of Vehicle-to-Everything (V2X) communication, drones, and healthcare. These application domains have a diverse set of requirements on QoS, where their application performance highly depends on meeting those networking requirements. Suppose the example of inter-vehicular communication with vehicular platooning as prime example: A leader vehicle informs its following vehilces periodically about its current speed and acceleration/deceleration, allowing two ore multiple vehicles to form a platoon in order to increase the road traffic efficincy. Whenever the communication between the leader vehicle and its follower vehicles (and between the other vehicles in the platoon) breakes, a save operation of vehicular platooning is not possible any more which could lead to severe crashes in the worst case. Yet, a complete breakdown of communication is not necessary to lower the performance of the vehicular networking application, already slight changes in the latency of transmitted information leads to severe problems for the application, and thus prevents save operation of vehicular platooning. In that context, novel networking protocols come into play which deal with varying and unpredictable networking conditions, e.g., by using heterogeneous networking, or increasing the safety distance whenever networking performance is low.

A study has been conducted by Erunkulu et al. [4] inspecting the required channel characteristics of 5G that would suit various applications discussed by the authors. Detailed surveys pertaining to usage of 5G in individual application domains like vehicular communication and industrial automation have also been carried out in the works of Alalewi et al. [5] and Gangakhedkar et al. [6]. While these studies show that 5G is a potential enabling technology in different domain verticals, it has to be adapted to overcome challenges of strict upper bound of latency and throughput for reliability. These adaptions need to be followed by thorough testing of the 5G network system where the network performance critical metric values can be reproduced.

A core commonality of many to all use cases for network communication in cyber-physical systems (e.g., vehicular networks) is that they need a proper reproducible performance evaluation under varying networking conditions. In context of those performance evaluations of applications, the state-of-the-art lists many approaches to achieve that tasks, e.g., work from Esmaeily et al. [7] which presents a cloud-based solution that facilitates testing 5G with network slicing and incorporates Software Defined Networking (SDN) and Network Function Virtualization (NFV).

The network can be configured to obtain multiple QoS profiles based on different traffic flows, which makes it crucial to measure and reproduce the network characteristics of these traffic patterns. In that context, recent works have investigated those challenges. He et al. [8] proposes a distributed test framework for 5G network which is implemented with Remote Procedure Calls (RPC) in a client-server based architecture for evaluation of the network elements. An increased number of services are required to run as this system is scaled. As such, a distributed architecture with centralized control for services is introduced by Chai et al. [9]. Their architecture supports the testing of individual components of a 5G network, e.g., their interfaces, but without addressing end-to-end performance of 5G. In [10], pertaining to testing of integrated modular avionics, a distributed testing framework has been proposed. The architecture therein is deployed with a centralized control module and measurement server endpoint which executes on a single entity. This restricts conducting performance evaluation in parallel in a network system implemented with multiple services.

Due to the complex interplay of all networking elements in a 5G network, it is often impossible to have a representative digital twin of the overall network which allows to reproduce the exact networking conditions. Therefore, various research works have been carried out to emulate the 5G network based on an abstracted model and view of the 5G system. In the work of Chirivella-Perez et al. [11], a cost-effective emulator is presented that uses commercial off-the-shelf devices. Ostinelli et al. [12] worked on a container-based approach to build a 5G emulator. Emulation based on Software Defined Radio (SDR)-based 5G networks has been presented in [13]. Their emulation framework is built on container based solution which allows the modules of the emulation framework to be re-usable and scalable. But, the framework focuses on specific use cases involving ultra-dense and hotspot area use-cases. The scalability of all the above presented frameworks are realised using virtualisation. This produces a challenge in reproducing network characteristics introduced by random or non-determininstic events using real hardware during tests on the network. From a simulation perspective, the OMNeT++ discrete event simulator [14] can integrate a 5G simulating framework known as Simu5G [15]. Going beyond simulation, Nardini et al. [16] designed a 5G emulator based on Simu5G that reproduces the network channel characteristics in real-time. It recommends choices of modelling to build and scale the realised emulator. However, the emulation is not based on real-world measurements and therefore can not factor in certain anomalies, such as interference caused by a neighboring 5G network or device malfunctions. In order to reproduce the real behavior of the network under test, it is mandatory to perform actual measurements and replay the collected metrics later using network impairment. In context of network impairment techniques, Schütz et al. [17] presented an open-source link emulation bridge called *Link'em*. It extends on the GNU/Linux based *tc-NetEm* for emulation and enables use of sophisticated packet loss models along with existing *tc-NetEm* network impairments. Nokia's Extended Reality Lab in Spain developed a configurable 5G Radio Access Network emulation framework known as *FikoRE*[1]. It has been showcased to be highly modular and hence allow for easy modifications. It can be used as a simulator and emulator for various application use cases. The main objective was to study the behavior of network for specific application use-cases.

A main commonalty of those approaches is that they either investigate accurate modelling of 5G networks from a wireless channel perspective, or focus on emulation methodologies of network traffic. In our paper, we close this gap of joint measurement and emulation by bridging pure network emulation based systems with systems to elaborate the networking characteristics from an application point of view.

The *DecMEN* framework presented in this work defines its architecture in a distributed way, allowing for both isolated end-to-end measurements in the test network as well as for the existence of a mobile Measurement Client entity. It can be deployed on real hardware systems to reproduce the real-world 5G network measured channel characteristics. Essentially, *DecMEN* measures the network characteristics of a 5G network in the first phase, and reproduces the measured network characteristics in a controlled environment in the second phase. The proposed framework architecture can be scaled as required by use cases with measurement entities existing at the endpoints to ensure end-to-end measurement of the network performance. Hence, our framework can measure, monitor and emulate end-to-end channel characteristics of a 5G network as a whole. The emulation methodology in our framework allows for reproducing the channel characteristics for detailed and extended tests required on applications across domains. The entire operation of our *DecMEN* framework is coordinated from a web-based dashboard asynchronously. The proposed *DecMEN* framework rests on and is built upon our previous prototype presented in [18]. The core contribution of this work is to design and develop an impairment technique to emulate the measured network data rate and latency. We compare the measured network characteristics to the characteristics emulated by *DecMEN* in order to assess its accuracy.

III. Measurement Framework Architecture

Building upon our previous publication [18] which facilitates the measurement process to obtain the quality of a 5G network, we outline in Figure 1 the architecture of our system. We adapted the control communication to facilitate the *WebSocket* protocol, improving the resilience while reducing delays of the control instructions. We are utilizing a client-server architecture performing end-to-end measurements which are coordinated by an *Orchestrator*. As the Measurement Client is a mobile device and can be moved to strategically relevant locations, our system allow us to collect spatio-temporal samples of measurements, marking data with externally acquired location information.

[1]https://github.com/nokia/5g-network-emulator

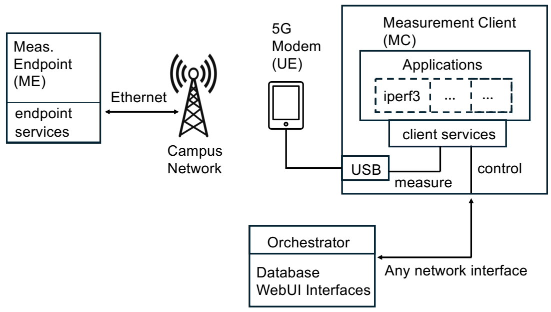

Figure 1. Measurement system architecture representing the *Orchestrator*, *Measurement Client* and *Measurement Endpoint* with the 5G Campus Network under test.

A. Orchestrator

We use Out-of-Band (OOB) communication to handle all control instructions with different entities in *DecMEN* to prevent transmission of control signals over the test network, avoiding any potential falsification of measurement results. Measurement data is stored persistently in an SQL-based database. The *Orchestrator* hosts a web user interface as a dashboard for *DecMEN* from which we start and stop network measurements and visualise the results.

B. Measurement Client

The Measurement Client receives requests from the Orchestrator through the web user interface to start measurements in the test network. The Measurement Client then sends the measurement results to the Orchestrator. Since the Measurement Client is a mobile device in our framework, it enables collection of spatio-temporal data points. Our framework records the measurement metrics once every second. To achieve scalability of the system, multiple Measurement Clients and the same number of Measurement Endpoints can be used simultaneously.

C. Measurement Endpoint

The Measurement Endpoint is connected to the core of the test 5G network. This enables an end-to-end performance evaluation. It serves as the counterpart of the Measurement Client and hosts the *iperf3* server.

IV. Emulation Architecture

In Figure 2, we depict the network emulation architecture of our system. As stated above, the emulation network utilizes the same entities namely Orchestrator, Measurement Client and Measurement Endpoint. Additionally, an Impairment Box is introduced between the Measurement Client and Measurement Endpoint. It enables a connection between Measurement Client and Measurement Endpoint by creating a transparent network bridge. An Ethernet connection is used to connect from Measurement Client to the Impairment Box and from the Impairment Box to the Measurement Endpoint. To impair the network between Measurement Client and Measurement

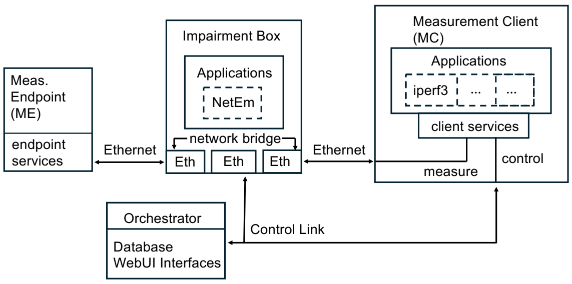

Figure 2. Emulation system architecture representing the *Orchestrator*, *Measurement Client*, *Impairment Box* and *Measurement Endpoint* with the emulated network under test.

Endpoint, we are using *tc-NetEm* from GNU/Linux operating at the two member interfaces of the bridge. It consists of a queue discipline also known as *qdisc* to manage a configured network impairment. *tc-NetEm* attaches this queue discipline to the network interface on which a specified impairment is requested. This is a virtual queue applied on a given interface that can manipulate the network traffic according to the requested impairment parameters. It enables one or more network impairments such as maximum data rate, communication latency, and more.

In our proposed measurement framework, we first measure throughput and latency in our 5G Campus network. Afterwards, we use the measured metrics as constraint values for the emulation: Our Impairment Box restricts the throughput to be at most as high as the measured value. Also, the measured latency is added to each packet transmitted through the Impairment Box. This way, the emulated network is supposed to show similar characteristics to the measured network. In order to quantify how accurate the emulation actually is, we do the following: Firstly, we conduct measurements, retrieving throughput and latency values each second. Secondly, we use this data as input parameters for the emulated network. Thirdly, we measure now the impaired network: We connect our Measurement Endpoint and Measurement Client over the Impairment Box. We start new measurements over the now impaired network and retrieve again the throughput and latency values. For every real-world measurement, we conduct ten measurements in the emulated network in order to avoid falsifying measurements by changing channel conditions. This results in the measurement of the emulated network requiring ten times as much time as the original, real-world measurement. Finally, we compare the original measured values of the real network retrieved in the first step to the measured values of the impaired network in order to quantify the accuracy of the emulation.

V. Prototypical Implementation

To show the feasibility of our approach, we implemented the three entities (*Orchestrator*, *Measurement Client*, *Measurement Endpoint*) on Raspberry Pis. For simplicity, we use the onboard WLAN modules of the devices for OOB control communication. The control instructions via OOB are exchanged using the *WebSocket* protocol. It utilises TCP mode of network traffic for communication. *WebSocket* uses frame headers to identify the frame sizes and the frames containing the payload message. The *WebSocket* API rearranges the TCP pieces of data into frames that are collated into messages. The Measurement Client is connected via a smartphone by using USB tethering to the 5G campus network. Further, the Measurement Endpoint is connected via ethernet to the 5G Core network under test. To ensure correct routing of traffic among the 5G network elements and our OOB communication, we use GNU/Linux network namespaces to separate the two communication links. A completely isolated network stack is initiated upon the creation of a new network namespace. GNU/Linux network namespaces enable the maintenance of separate IP addresses, routing tables and configurations. Hence, it allows for existence of multiple independent network stacks within a single system. It also separates any processes operating within a given network namespace such that processes belonging to different namespaces cannot communicate with each other.

To build the emulation system, we use an APU 4 from PC Engines consisting of a 4-core CPU with 4 GB of DRAM and being compatible with x86_64 CPU architecture. We use GNU/Linux (Debian 12) containing *tc-NetEm* to enable network impairment by using a transparent software bridge for network communication between two Ethernet ports.

To show the effectiveness of our joint measurement and emulation approach, we measure and emulate the throughput and latency of our private 5G Campus Network within our research group. Due to space constraints we are not focusing on the tooling aspects of the system in the remainder of this section.

To quantify the data rate and the latency of the communication, we are using the widely utilized tools *iperf3*[2] and *ping*. *iperf3* is configured to send 100 Mbit/s of data with a message payload size of 1450 Byte. We parameterise it with '-R' to configure a downlink traffic such that the Measurement Endpoint transmits data to Measurement Client. The results are sent periodically via OOB communication through the *WebSocket* protocol to the Orchestrator, which stores them in the database.

VI. Preliminary Results

To assess the performance of our *DecMEN* framework and validate its operation in realistic situations, we perform measurements with it in our 5G campus network and compare it to the performance of the network under the influence of emulation. Essentially, we perform two sets of tests to compare prime networking metrics, namely the achievable received throughput and communication latency. We measure the data rate in UDP mode and latency via *ICMP*. To avoid negative impact of inaccuracies of time synchronization when measuring the one-way latencies of communication, we report results from latency measurements as round-trip times. In order to obtain statistical significance of the gathered data, we ensure that the measurements of the 5G networks last at least 30 min of time

[2]https://github.com/esnet/iperf

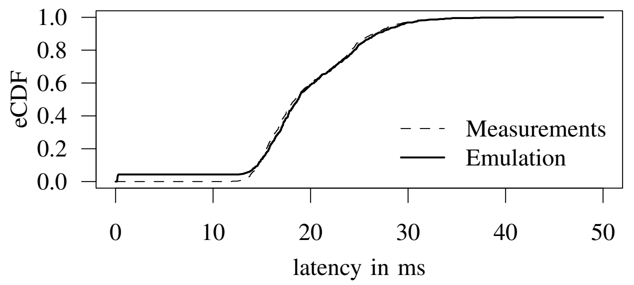

Figure 3. eCDF plot to depict the distribution of emulated latency and measured latency.

duration. This is done because each measured metric value in the 5G campus network corresponds to ten emulated metric values in the impaired network as described in Section IV.

For our feasibility study, we configure our 5G campus network with an uplink to downlink ratio of 5 : 5, operating at the 3750 MHz band by taking advantage of 100 MHz channel bandwidth and using a transmit power of 23 dBm. To avoid corner effects, we perform our feasibility study with a single active 5G base station in an isolated and controlled environment.

Measurements reading zero for either throughput or latency are filtered out. This is necessary to eliminate any anomalies that could affect the result and analysis specially for the latency tests. Further, given that each measured metric value in the 5G campus network corresponds to ten emulated metric values in the impaired network, a methodical approach is employed to facilitate a meaningful comparison. Specifically, we compute the average of ten data points measured from the emulated network for each metric. This aggregated value is then compared against each corresponding single data point measured from the 5G network. By using this structured methodology, the comparative evaluation ensures that the analysis is based on equivalent sets of data points. This provides a strong basis for assessing the network performance under varying conditions with an enhanced accuracy.

The test result for a comparative evaluation between measured and emulated latency in *DecMEN* framework through an eCDF plot is presented in Figure 3. The resultant plot shows that the emulated latency in our framework corresponds to the measured latency values in our 5G campus network.

In the measurement of the emulated test network, the emulation methodology was to measure the latency of the test network by modifying the link layer while network measurements are continuously carried out. This resulted in measurement of the test network while the Impairment Box in our framework remained engaged in setting up the link layer. Therefore, in the link layer of the Impairment Box, the existing queue discipline is removed and this is followed by addition of another queue discipline. There is a time interval in which none of the queue disciplines are applied resulting in measured latency values lower than that of 5G campus network. This behavior is depicted in Figure 3, where it can be seen that lower latency values in the emulated network occurs for a small amount of data samples (around 5 %) in comparison to those of the real 5G campus network. This behavior occurs before the emulated network attempts to match the measured latency values. Overall, the latency values in the emulated network match that of the test network as can be seen in Figure 3.

In order to adapt our framework to emulate the network characteristics better, we change the method of emulation in our framework. As the metric values to be emulated are received by the Impairment Box, it sets up the queue discipline of test network before starting the impairment and measurement of the impaired network. This prevents any measurement on the emulated network while the impairment is still under progress.

To verify the effectiveness of this improved approach, we perform a second campaign to compare and evaluate the performance of test network and emulated network.

Figure 4 shows the test result of a comparative performance evaluation of data rate in our 5G Campus network and the emulated network for this second measurement campaign, where we focus on the goodput of the 5G network. The network impairment is configured on the Data Link layer of the networking stack. This carries a significant difference from the measured throughput of a network since at Application Layer, the rate of new information (i.e., goodput) is being measured, while impairing at Link Layer also protocol headers will impact the achievable goodput. To quantify for this difference and validate the performance of network under *DecMEN* emulation, we also indicate and plot an expected emulation of data rate with respect to the measured data rates in Figure 4.The plot shows that *DecMEN* emulation performance aligns with the expected data rate of the goodput, which we outline in the next subsection in detail.

A. Analytical Evaluation of Goodput

To compute the expected data rate, all the overheads due to the UDP packet header sizes and the application payload volume must be considered. It is to be noted that the emulation - other than the measurement in the real 5G Campus network - is being performed over an Ethernet connection between Measurement Client and Measurement Endpoint with the network being impaired by a dedicated bridged Impairment Box. To investigate and verify the packet sizes being received over the emulated network, we captured frames at the downlink Ethernet interface of the Measurement Client and analyzed it. Since we use Ethernet II frame format, the total UDP packet size in our emulation consists of the UDP header (8 Byte), IP header (20 Byte) and ethernet header (14 Byte). This accounts for a total header size of 8 Byte + 20 Byte + 14 Byte = 42 Byte. Hence, the total size of a UDP packet in Byte is derived as (UDP Packet Payload Size + Total Header Size). We use the default UDP payload size provided by *iperf3* as 1450 Byte to ensure that IP fragmentation does not take place in our network. This results in a total packet size of 1450 Byte + 42 Byte = 1492 Byte per UDP packet. Hence, our framework takes a total UDP packet size of 1492 Byte incorporating all lower layer headers. We perform the tests using a target data rate of

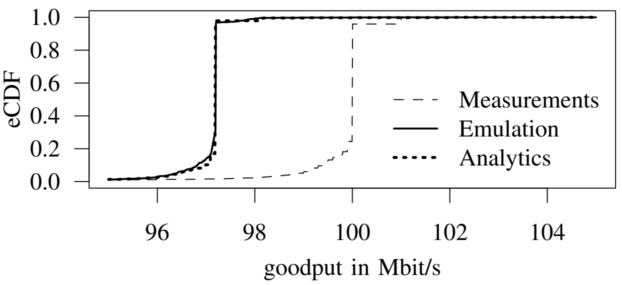

Figure 4. eCDF plot to depict the distribution of emulated throughput, measured throughput and analytical emulation in UDP mode.

100 Mbit/s in the Link Layer of the IP stack. The MTU at link level is set at 1500 Byte in our framework. The expected data rate is then computed by multiplying the ratio of application payload size and total packet size to measured bit rates of the 5G Campus network as given below.

$$\left(\frac{\text{Payload Size per Packet}}{\text{Total Packet Size with Headers}}\right) \times \text{Measured Throughput} \tag{1}$$

Finally, in order to plot the expected data rates, each data point obtained from the measured data rates during the measurement campaign in the 5G Campus network is utilized as input into the Equation (1). This procedure ensures that the empirical data collected is directly integrated into the analytical model. This allows for an accurate representation of the expected performance.

VII. Conclusion

In this work, we have presented our developed framework called *DecMEN* to measure a network and emulate the measured network characteristics of data rate and latency over a period of time. The framework is implemented with open-source tools and tested for evaluation of the framework. Upon performing a comparative evaluation of measurement and emulation of our 5G Campus network, our framework depicts that the data rate and latency matches. However, the emulated data rate has been shown to have a better alignment to the measured values as compared to that of latency. In future work, we will systematically investigate and test the scalability of our system by increasing the number of Measurement Clients and Measurement Endpoints.

VIII. Acknowledgement

This work was supported by the German Aerospace Center (DLR) and the Federal Office for Information Security (BSI) in context of the DeSiRe-NG project.

References

[1] ETSI, “3GPP TR 38.912 version 15.0.0 Release 15,” European Telecommunications Standards Institute, Tech. Rep., 2018.

[2] H. X. Nguyen, R. Trestian, D. To, and M. Tatipamula, “Digital twin for 5G and beyond,” *IEEE Communications Magazine*, vol. 59, no. 2, pp. 10–15, 2021.

[3] H. Ullah, N. Gopalakrishnan Nair, A. Moore, C. Nugent, P. Muschamp, and M. Cuevas, “5G Communication: An Overview of Vehicle-to-Everything, Drones, and Healthcare Use-Cases,” *IEEE Access*, vol. 7, pp. 37 251–37 268, 2019.

[4] O. O. Erunkulu, A. M. Zungeru, C. K. Lebekwe, M. Mosalaosi, and J. M. Chuma, “5G Mobile Communication Applications: A Survey and Comparison of Use Cases,” *IEEE Access*, vol. 9, pp. 97 251–97 295, 2021.

[5] A. Alalewi, I. Dayoub, and S. Cherkaoui, “On 5G-V2X Use Cases and Enabling Technologies: A Comprehensive Survey,” *IEEE Access*, vol. 9, pp. 107 710–107 737, 2021.

[6] S. Gangakhedkar, H. Cao, A. R. Ali, K. Ganesan, M. Gharba, and J. Eichinger, “Use Cases, Requirements and Challenges of 5G Communication for Industrial Automation,” in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2018, pp. 1–6.

[7] A. Esmaeily, K. Kralevska, and D. Gligoroski, “A Cloud-based SDN/NFV Testbed for End-to-End Network Slicing in 4G/5G,” in *2020 6th IEEE Conference on Network Softwarization (NetSoft)*, 2020, pp. 29–35.

[8] 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. [Online]. Available: https://doi.org/10.1145/3603781.3603878.

[9] 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. [Online]. Available: https://doi.org/10.1145/3468691.3468698.

[10] W. Sheng, J. Yu, Z. Weng, and D. Tang, “A distributed and reusable workflow-based auto test framework for distributed integrated modular avionics,” in *IEEE AUTOTESTCON*, 2017, pp. 1–8.

[11] E. Chirivella-Perez, J. M. Alcaraz Calero, Q. Wang, and J. Gutiérrez-Aguado, “Towards a Realistic 5G Infrastructure Emulator for Experimental Service Deployment and Performance Evaluation,” in *2018 IEEE/ACM 22nd International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, 2018, pp. 1–7.

[12] N. K. Ostinelli, S. T. Arzo, F. Granelli, and M. Devetsikiotis, “Emulation of LTE/5G Over a Lightweight Open-Platform: Re-configuration Delay Analysis,” in *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021, pp. 01–06.

[13] A. Kretsis, P. Soumplis, P. Kokkinos, and E. A. Varvarigos, “An SDN Emulation Platform for Converged Fiber-Wireless 5G Networks,” in *2021 International Conference on Computer Communications and Networks (ICCCN)*, 2021, pp. 1–7.

[14] A. Varga and R. Hornig, “An Overview of the OMNeT++ Simulation Environment,” in *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops*, ser. Simutools ’08, Marseille, France: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[15] 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.

[16] 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.

[17] B. Schütz, S. Thieme, N. Aschenbruck, et al., “Link ’em: An Open Source Link Emulation Bridge for Reproducible Networking Research,” in *2019 International Conference on Networked Systems (NetSys)*, 2019, pp. 1–3.

[18] N. Mondal, M. Niggemeyer, S. Welzel, and F. Klingler, “Poster: Decentralized Framework for Measuring End-to-End Performance in 5G Networks,” in *15th IEEE Vehicular Networking Conference (VNC 2024)*, to appear, Kobe, Japan: IEEE, May 2024.