Small-Scale Testbed for Evaluating C-V2X Applications on 5G Cellular Networks

Kaj Munhoz Arfvidsson[*,1], Kleio Fragkedaki[*,1], Frank J. Jiang[*,1], Vandana Narri[1,2],
Hans-Cristian Lindh[3], Karl H. Johansson[1], Jonas Mårtensson[1]

Abstract— In this work, we present a small-scale testbed for evaluating the real-life performance of cellular V2X (C-V2X) applications on 5G cellular networks. Despite the growing interest and rapid technology development for V2X applications, researchers still struggle to prototype V2X applications with real wireless networks, hardware, and software in the loop in a controlled environment. To help alleviate this challenge, we present a testbed designed to accelerate development and evaluation of C-V2X applications on 5G cellular networks. By including a small-scale vehicle platform into the testbed design, we significantly reduce the time and effort required to test new C-V2X applications on 5G cellular networks. With a focus around the integration of small-scale vehicle platforms, we detail the design decisions behind the full software and hardware setup of commonly needed intelligent transport system agents (e.g. sensors, servers, vehicles). Moreover, to showcase the testbed's capability to produce industrially-relevant, real world performance evaluations, we present an evaluation of a simple test case inspired from shared situational awareness. Finally, we discuss the upcoming use of the testbed for evaluating 5G cellular network-based shared situational awareness and other C-V2X applications.

I. INTRODUCTION

Over the recent years, the interest in exploring the use of vehicle-to-everything (V2X) communication has grown immensely. Due to factors like unprotected road-users and occlusions, there are complex traffic scenarios that are difficult for automated vehicles to resolve individually. To safely and efficiently handle these scenarios, automated vehicles need to communicate with each other, infrastructure, and central servers. This has led to the development of various Intelligent Transport Systems (ITS) that utilize wireless technologies including different varieties of dedicated short-range, direct communication, 4G, and 5G for V2X communication to enhance traffic safety and improve road

Fig. 1. A snapshot of the 1/10th-scale SVEA platform driving in the presented 5G-based C-V2X testbed in the Kista Innovation Park. More details about the SVEA platform at [https://svea.eecs.kth.se]

throughput [1]. Vehicles that utilize V2X communication to enhance their automation are referred to as connected and automated vehicles (CAV). Moreover, throughout this work, we refer to the collection of ITS services that leverage V2X communication as V2X applications.

Already today, there are a variety of V2X applications that have been developed using various wireless technologies. These V2X applications are typically add-on services to the core functionality of CAVs that leverage communicated data to implement functionalities like beyond line-of-sight collision warnings, intersection movement assistance, or left turn assistance [2], [3]. V2X applications are developed to further enhance safety and efficiency in scenarios where the sensor suite onboard the CAV is unable to perceive critically important information. One example of this is shared situational awareness, where sensors, servers, and vehicles share their perception data with each other using V2X direct communication to enhance each other's situational awareness [4]. V2X applications are predominantly developed around one of two main V2X communication standards: ITS-G5 [5] and C-V2X [6]. ITS-G5 is based on Wi-Fi communication on the 5.9 GHz band. C-V2X is an umbrella technology which encapsulates all 3GPP V2X technologies, including both direct communication and cellular network communications [7]. While there are some preliminary studies performed in [8] on C-V2X using 5G cellular networks, many of the studies

conducted so far are developed around using 4G. We attribute this to the current difficulty around prototyping advanced C-V2X applications in a controlled environment with a real 5G network, hardware, and software in the loop.

To help alleviate this difficulty, we design and develop a testbed specifically for evaluating the real-life performance of advanced C-V2X applications on 5G cellular networks. More specifically, we design a testbed to enable low-cost experimentation on advanced C-V2X applications with the ability to precisely evaluate the 5G cellular network, hardware, and software's performance. To make experimentation cheaper, we build the testbed around the use of a 1/10th-scale CAV platform. Although 1/10th-scale CAVs do not capture the full dynamics of full-scale CAVs, they do provide the ability to cheaply evaluate V2X applications with real network, hardware, and software in the loop [9]. Additionally, 1/10th-scale CAVs offer the ability to do preliminary studies with motion, providing some initial insight into how the results translate to full-scale CAVs.

A. Contribution

In summary, the main contribution of this paper is the design of a testbed for rapidly evaluating the real-life performance of C-V2X applications on 5G cellular networks. Explicitly, our contributions are three-fold:

1) we overview the design choices behind the network configuration, software, and hardware for supporting small-scale evaluations of the real-life performance of C-V2X applications on 5G cellular networks,
2) we showcase the testbed's capability to measure the real-life performance of a simple, situational awareness-inspired, test case under the realistic conditions of: clock synchronization error, network traffic overloading, and base station handovers,
3) we discuss the implications from the test case on the future evaluation of 5G-based C-V2X applications.

Additionally, the software used for sending C-V2X data over the 5G cellular network is available publicly.[1]

II. Motivation

In this section, we elaborate on the motivation for developing the testbed presented in this paper. Despite the significant effort and work that has previously been put into proposing V2X solutions, researchers still struggle to develop and evaluate the end-to-end performance of V2X applications with a real hardware in the loop. Given the relatively recent introduction of 5G cellular networks, this difficulty is especially apparent. To this end, the motivating challenges we address in this work are the following:

1) how do we develop a modular testbed where 5G cellular network-based C-V2X applications can be implemented with less lead time for testing and evaluation?
2) what developments are necessary to replicate scenarios with 1/10th-scale vehicles under realistic conditions?

To address the first challenge, we go through the design decisions that make it easier to develop and test C-V2X applications on 5G cellular networks. These design decisions are carefully made while considering the trade-offs between development speed and resultant performance. Throughout the work, we note the trade-offs of our design decisions and mention alternatives that could improve performance with the trade-off of increasing implementation difficulty.

To address the second challenge, we implement a simplified test case inspired by the requirements of shared situational awareness. In this test case, we adapt the architecture in [4] to better suit 5G cellular networks and implement a generic version of the messaging required to support these types of applications. We test this adapted architecture under the realistic conditions of clock synchronization error, network traffic overloading, and base station hand-over events. While we draw inspiration from the implementations done in [4], we have chosen not to implement the full shared situational awareness application so we can obtain results that are more relatable to other C-V2X applications. Moreover, since the implementation is done with 1/10th-scale vehicles, the available software and hardware that can be integrated greatly differs from full-scale vehicles. At 1/10th-scale, we need to work with hardware and software more typical in robotics and mobile device development.

III. Testbed Design

In this section, we present our testbed (illustrated in Fig. 2) for 5G-cellular network-based C-V2X applications and how it can evaluate end-to-end performance. In particular, we detail the important design decisions behind the dedicated 5G standalone test network, ITS agent implementations, messaging system, and clock synchronization software in our testbed. For simplicity, we refer to 5G-cellular network-based C-V2X as "5G C-V2X" for the remainder of the paper.

A. Dedicated 5G Standalone Test Network

To fully evaluate a 5G C-V2X application, it is necessary to recreate real-life conditions that impact the cellular network. Central to our testbed is a dedicated 5G standalone test network that allows for custom network configuration. With it, we can precisely modify background load and enable features that protect traffic, ensuring a realistic setting for experiments. It is important to investigate the trade-offs related to network performance as it is not only influenced by the traffic volume but also by the number of users, the direction of the traffic, and the configuration of the network. Our testbed's network has a 20 MHz bandwidth and uses a Time Division Duplex (TDD) pattern `DDDSU`, where `D` represents a downlink slot, `U` represents an uplink slot, and `S` represents a special slot with 10 downlink symbols and 2 uplink symbols. Furthermore, a radio cell, in this network, has maximum capacity of ca. 40 Mbps. Lastly, in the context of our scenario, we utilize a specific QoS mechanism in the 5G radio access network called absolute priority scheduling.
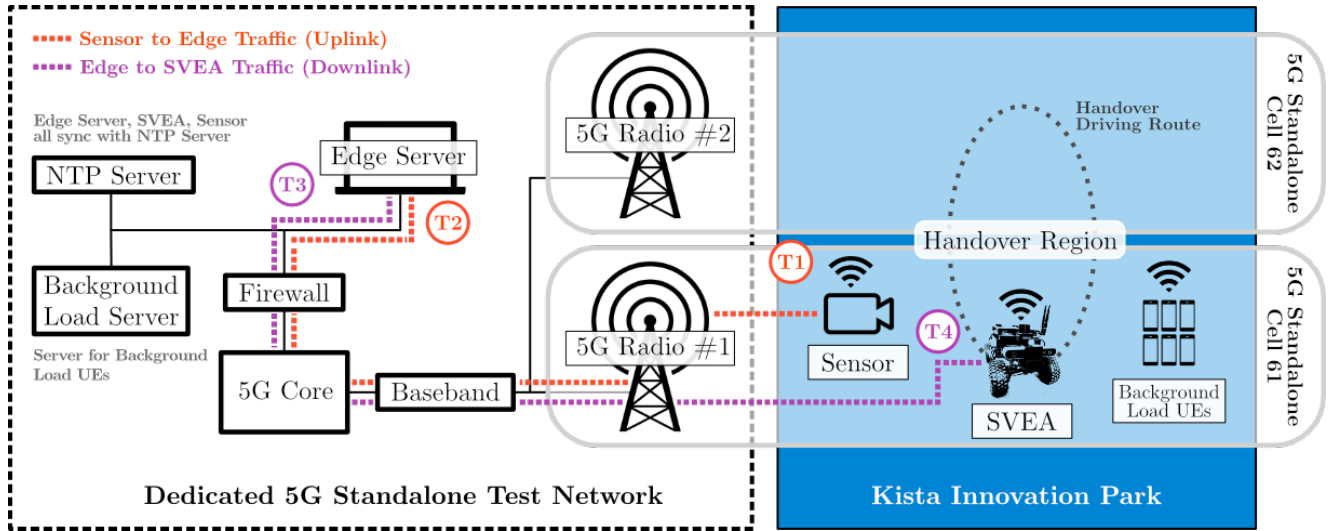
Fig. 2. Architecture and Spatial Distribution of our 5G C-V2X Testbed System. There are three types of ITS agents: infrastructure sensor, edge server and CAVs (SVEA). To communicate, the sensor and CAVs have a wireless connection using a 5G cellular network with 20 MHz bandwidth. Each radio cell has ca. 40 Mbps capacity. T1, T2, T3, T4 are the transmission time stamps of the messages used in our evaluation. All agents rely on an NTP server for clock synchronization. The NTP server is reached using the same wired or wireless connection as is used for their main communication.

B. ITS Agents

For quick and low-cost evaluation of 5G C-V2X, the three agents (sensor, edge server, CAV/SVEA) shown in Fig. 2 are actualized with small-scale platforms. They implement their 5G C-V2X applications using the Robot Operating System (ROS) [10]. The use of ROS yields significant development benefits, as it provides wide, open-source support for most of the sensor and message types that are required in C-V2X applications. For example, in [11], authors contribute an open-source package containing ROS support for ETSI ITS messages for V2X communication. However, since ROS is a best-effort implementation, it is not the most suitable middleware for external communication between agents on the 5G cellular network. Thus, we only use ROS on the local networks of each ITS agent. External communication can instead be facilitated by messaging systems such as MQTT [12], NATS [13] or Zenoh [14]. These have a direct impact on the overall 5G C-V2X performance, and the choice often depends on various design aspects like application compatibility and preferred implementation language. In line with the objectives of this platform, to accelerate development and evaluation of 5G C-V2X applications, we use the Transmission Control Protocol (TCP)-based messaging system NATS with the NATS broker sitting on the edge server in Fig. 2. Since NATS supports distributed brokers, it enables high-performing implementations of edge architectures for V2X applications.

1) Sensor: The sensor in our 5G C-V2X testbed is designed to be application-agnostic. It represents any device that produces data within the network but does not consume it. This can characterize a wide range of different applications and concepts, such as local infrastructure sensors or by-stander CAVs. For external communication, it is reasonable to assume that an infrastructure sensor has a wired connec-

tion. However, in certain use cases or scenarios, a sensor may require the flexibility of a wireless connection. For this reason, in this work, the sensor is communicating with the other agents through the cellular 5G network. Ultimately, this leads to easier, more agile setups for outdoor experiments. However, if latency and reliability need to be minimized, a wired connection can be used instead.

2) Edge Server: The edge server has the role of a centralized computing unit that aggregates and processes information and may also engage in decision-making processes. In our testbed, it acts as a proxy between the infrastructure sensor and the CAVs. Any computation that should be done on the server can be emulated by delaying messages. Depending on the scenario and application, the server can be located locally or more centrally. By being physically close to its users, the server can deliver benefits such as reduced latency and higher bandwidth. This is enabled by the 5G network's ability to deploy applications near the "edge" of an access network. However, it may be tolerable in some applications to instead deploy in far-away data centers, which can be more convenient or provide access to more computational hardware. Since we are prioritizing latency and bandwidth in our applications, we deploy the server on the edge in our testbed setup.

3) 1/10th-scale CAV (SVEA): The vehicle has been developed as a stand-alone platform for evaluating 5G C-V2X use cases. Introduced in [9], the SVEA is a 1/10th-scale automated vehicle that consists of the necessary components, both in hardware and software, to build an autonomous system. This enables the platform to evaluate a wide range of scaled-down traffic scenarios. Moreover, the platform can be used to collect and log vehicle state data, such as velocity and GPS position, even when driving manually. This makes the platform ideal for network-centric experiments that do not require any autonomy, but where mobility still plays an

151

important role. In contrast to the sensor, in the test case we set up, the SVEA represents a user in the network that only consumes data.

C. Messaging

In our testbed, communication between the agents occurs in both uplink and downlink channels. As illustrated in Fig. 2, the infrastructure sensor sends messages to the edge server, which then distributes those messages to the vehicle. We name the connections after the primary direction of the application-critical traffic. That is, UL for uplink from sensor to server and DL for downlink from server to vehicle. However, due to TCP, which requires acknowledgments to be sent in the opposite direction, there also exist downlink traffic from UL and uplink from DL.

The traffic is characterized by the transmitted messages. A message contains data of a certain size, and is sent at a certain rate. The contained data can be in any format, however, a message must include the timestamps, clock offsets, a sequence number, and a checksum to verify the message was sent correctly. This information will also later be used to determine latency. Before the sensor sends a message, it sets the T_1 timestamp from the system clock time. Once the message is received by the edge server, it updates the messages with a corresponding timestamp T_2. In the same way, when the edge server sends the message, and a vehicle receives it, then T_3 and T_4 are updated as depicted in Fig. 2.

The performance of 5G C-V2X applications can be evaluated with metrics such as latency, jitter, signal-to-noise ratio, packet drops and packet error rates. While the primary benefit of NATS is effortless deployment of edge architectures, since it is built on TCP, many of these performance metrics aggregates to latency due to TCP's order guarantees, error checks and retransmission algorithms. This simplifies the evaluation such that we can focus on measuring latency. For applications that need further latency minimization, technologies like MQTT can be used with UDP, which introduces the need to explicitly measure packet drops and packet error rates. However, by using MQTT with UDP, the implementation of edge architectures becomes more difficult since MQTT brokers cannot be directly distributed without additional implementations. As previously described, messages are sent in both UL and DL traffic. Therefore, we can also distinguish between L_{UL} and L_{DL} latencies. If all agents were perfectly synchronized, the latency would simply be the difference between the respective timestamps, e.g., $L_{\mathrm{UL}} = T_2 - T_1$.

D. Network Time Protocol for Synchronization

Measuring latency between different agents is difficult due to synchronization errors. Consider T_1 and T_2 which are measured on different clocks. This becomes a major source of errors when considering how computer clocks can drift non-deterministically depending on voltage fluctuations, temperature changes, system load, and production quality. Consequently, taking the time difference $T_2 - T_1$ would not only measure the communication latency, but also include any synchronization error between the clocks.

In our testbed all agents use Network Time Protocol (NTP) [15] to both correct their system clock and periodically query the NTP server for synchronization errors. By querying the NTP server during a test, we can correct for synchronization error in the latency measurement. As such, when setting timestamps in the messages, we also set a corresponding clock offset value. To account for the synchronization error, the latency becomes $L_{\mathrm{UL}} = (T_2 + E_2) - (T_1 + E_1)$.

While alternative methods like GPS Clocks or Precision Time Protocol (PTP) [16] can be utilized for more precise time synchronization, we have excluded them from our design due to hardware restrictions. Specifically, since server implementations are typically installed indoors, there is not always infrastructure available to provide servers access to a GPS receiver. Furthermore, we do not always have access to network infrastructures with hardware that has PTP support throughout. In the case of software-based PTP, the benefit over NTP is not as apparent when in a wireless network due to PTP being more sensitive to delay variations. Therefore, we choose to use the NTP client chrony[2] for clock synchronization due to it being readily available on most operating systems.

IV. Testbed Evaluation

In this section, we develop and evaluate a simple test case to assess the testbed's ability to support 5G C-V2X applications under various conditions. Specifically, we present the test case's latency under three distinct conditions that could impact the overall efficiency or reliability of 5G C-V2X applications. Our primary objective in this section is to show how the testbed can be used to evaluate the real-life performance of a 5G C-V2X application. To this end, we start by contextualizing our evaluation in a previously developed V2X application and use that application to determine a relevant architecture for us to test.

A. Test Case Context: Shared Situational Awareness

For our test case, we are inspired by [4], where the authors have implemented a shared situational awareness framework, with an objective to improve situational awareness of the ego-vehicle using V2X communication. In this cited paper, a simple urban scenario is considered, where the field-of-view of the ego-vehicle has an occlusion created by a parked vehicle. But this occluded region is visible for two other connected V2X units in the near surrounding. The connected V2X unit shares their perception data with the ego-vehicle.

The authors utilize the standardized Collaborative Perception Message (CPM) [17]. In this standard, a general message structure consists of Management Container (MC), Station Data Container (SDC), Sensor Information Container (SIC), Perceived Object Container (POC), and Free Space Addendum (FSA). In [4], the communication of CPMs is carried out over an ITS-G5 access layer with a 10 MHz bandwidth. Each message is sent and received at a frequency of 10 Hz. The total payload of each message was 156 bytes

---

[2]https://chrony-project.org/

and each message consists of MC, SIC, and POC. In each message, there were five POCs, which had a payload of 78 bytes. According to IEEE 802.11 standards, the ITS-G5 network has a limitation to send each message with a maximum payload of 1500 bytes. This means that if messages need to be sent with payloads more than 1500 bytes, the messages need to be sent in separate parts. Based on these values, we are able to determine some of the relevant message sizes and frequencies to include in our test case.

In this work, we evaluate our testbed with a simple test case that evaluates the messaging capabilities within the 5G cellular network to develop a preliminary understanding of how a 5G C-V2X shared situational awareness system could be designed. Since 5G networks have different bandwidth alternatives and different features for prioritizing sessions in the network, we set up a test case where we evaluate the testbed's performance with varying network loads, message sizes, and message frequencies. Moreover, since safety-critical data is being communicated, it's of great interest to understand the effects of network traffic overloading and handovers on the messaging between agents. As the focus of the developed testbed is the facilitation of development and evaluation of advanced 5G C-V2X applications, we will not focus on comparing the performance of 5G C-V2X with ITS-G5-based V2X. Instead, through our test, we are interested in understanding how shared situational awareness would look as a 5G C-V2X application.

B. Test Case Setup

Our testbed is in the Kista Innovation Park (KIP), an arena that provides a controlled environment for conducting research and development of C-V2X applications on a dedicated 5G standalone test network. As illustrated in Fig. 2, KIP is equipped with two radio stations, allowing for the study of handover events in a controlled manner. This, combined with the ability to freely configure its net-
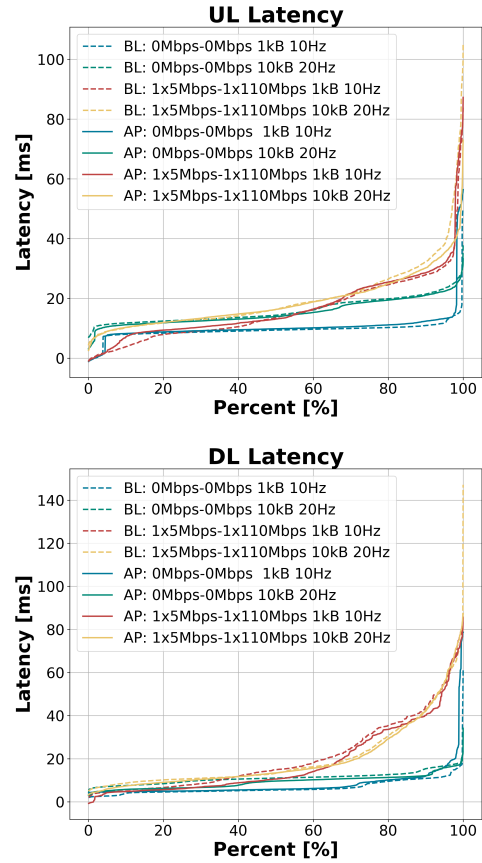

UL Latency

DL Latency

Fig. 3. Cumulative Distribution Functions (CDFs) of latency for all the different combinations under nominal conditions.

work, positions KIP as a suitable arena to explore real-world complexities, yet under controlled and manageable circumstances.

Based on the urban scenario introduced for shared situational awareness, for our test case, we employ an infrastructure sensor, an edge server, and a 1/10th-scale CAV as the primary agents. By setting up this type of test case, we can measure the latency of safety critical perception data when sent from a sensor to a CAV through an edge server over a 5G cellular network, as shown in Fig. 2. We can also compare the effect of different network configurations. In this work, we show this capability using a QoS mechanism in the 5G network named absolute priority scheduling that aims to protect our safety-critical traffic. We compare the absolute priority (AP) configuration with a baseline (BL) that uses default scheduling.

In our implementation, we send messages containing dummy data instead of real perception messages to more easily control certain traffic characteristics such as message size and frequency. The different combinations of network settings, background loads, message sizes, and frequencies applied in our experiments are summarized in Table I. With these, we show how flexible our testbed is in terms of how

	Network Config.	Background Load Uplink – Downlink [Mbps]	Message Size [kB]	Message Frequency [Hz]
Nominal	BL	No Load – No Load	1	10
			10	20
		1 UEs x 5 – 1 UEs x 110	1	10
			10	20
	AP	No Load – No Load	1	10
			10	20
		1 UEs x 5 – 1 UEs x 110	1	10
			10	20
Overload	BL	1 UEs x 40 – No Load	10	20
		2 UEs x 40 – No Load	10	20
	AP	1 UEs x 40 – No Load	10	20
		2 UEs x 40 – No Load	10	20
Mobility	BL	No Load – No Load	10	20

TABLE I
COMBINATIONS OF NETWORK CONFIGURATIONS, BACKGROUND LOAD, MESSAGE SIZES, AND FREQUENCIES IN THE PRESENTED TEST CASE.

quickly one can develop V2X scenarios and start testing different designs of 5G C-V2X applications. By exposing these options, we can create test cases relatable to existing ITS-G5 based standards, i.e. 1 kB messages at 10 Hz, then continue to investigate other test cases that take better advantage of the features in 5G networks. We note that these test cases are not exhaustive and in the future it'd be interesting to continue to test shared situational awareness under other scenarios. Next, we describe the specific network conditions we considered in this work.

1) Nominal Conditions: In the nominal conditions, we consider a static 1/10th-scale CAV that is receiving messages of a particular size and at a particular frequency. At the same time, there is background load in the network generated by regular User Equipment (UE) using iPerf [18], and this load is carried over User Datagram Protocol (UDP). Although the application data is carried over TCP through NATS, the background load has to flood the network, making UDP a more suitable transport protocol. For the 5G network, that has 20 MHz bandwidth, we consider 1 UE loading 5 Mbps in the uplink direction and 1 UE loading 110 Mbps in the downlink direction to be within nominal conditions with respect to the network's capacity.

2) Overload Conditions: In the overload conditions, we are also analyzing conditions when there is background load exceeding the network's capacity, which we refer to as overloading conditions. Since networks are typically not configured for symmetric resource allocation in uplink and downlink traffic, it can be further valuable to test background load in each direction separately. To assess the potential benefits of prioritized network traffic on 5G C-V2X performance, we test the network configurations AP and BL when there is asymmetric background load.

3) Mobility Conditions: In the mobility conditions, due to the motion of the vehicle, handover events are triggered when the vehicle transitions between the two radio cells, as illustrated in Fig. 2. These events can increase latency, which is why they are an important aspect to consider during mobility conditions.

C. Test Case under Nominal Conditions

Under nominal conditions, in a static vehicle agent setting, we examine our test case with varying message types, background load, and network configurations, as detailed in
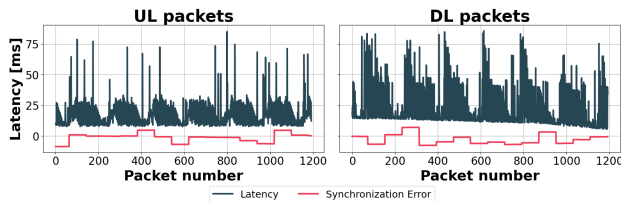
Fig. 4. Latency of individual packets under background load. This test was conducted with message size and frequency of 1 kB at 10 Hz, in BL network configuration, while there was 1 UEs x 5 Mbps uplink and 1 UEs x 110 Mbps downlink load. In red, we see the synchronization error between sensor-server and server-vehicle, respectively.
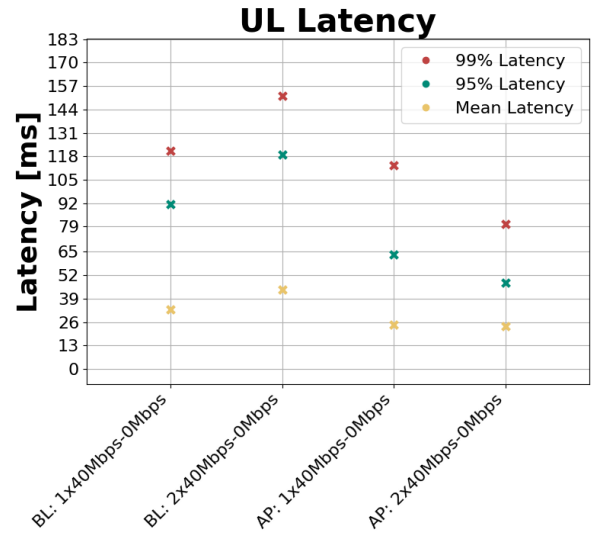
Fig. 5. Comparison between different cases under overload conditions. The figure depicts 99%, 95%, and mean latency when overloading the network in uplink traffic.

Table I. Using Cumulative Distribution Functions (CDFs) of latency, we can compare the impact of different traffic and network attributes with respect to the given requirements of a 5G C-V2X application. For example, a requirement could be that 95% of the latency distribution must be less than 40 ms. Fig. 3 shows such CDFs for the test cases under nominal conditions. As expected, there is no significant difference between BL and AP network configurations, since the benefit of absolute priority scheduling is not apparent with such low load and just one background UE in the cell.

In addition to Fig. 3, it can also be useful to view the latencies of individual packets. Since our testbed logs each message, it is possible to perform such analyses, as depicted in Fig. 4. It is worth noting that this particular test was affected by synchronization errors due to drifting system clocks. Depending on what is being tested, this might not be a significant problem. However, this issue demonstrates the importance of careful consideration of, and planning for, time synchronization.

D. Test Case under Overload Conditions

Overload conditions, as the name conveys, describe the case when there is background traffic exceeding the network's capacity. In Fig. 5, we present the 99%, 95%, and mean latency values of the UL traffic when there is extreme background load only in the uplink direction. Through these measurements, we see that the network was able to protect the UL traffic under overload conditions with the help of the enabled 5G features in AP.

E. Test Case under Mobility Conditions

Evaluating network behavior in the context of a moving vehicle is particularly interesting due to the handover events that can occur. The network configuration, message size, and message frequency of this experiment are described in
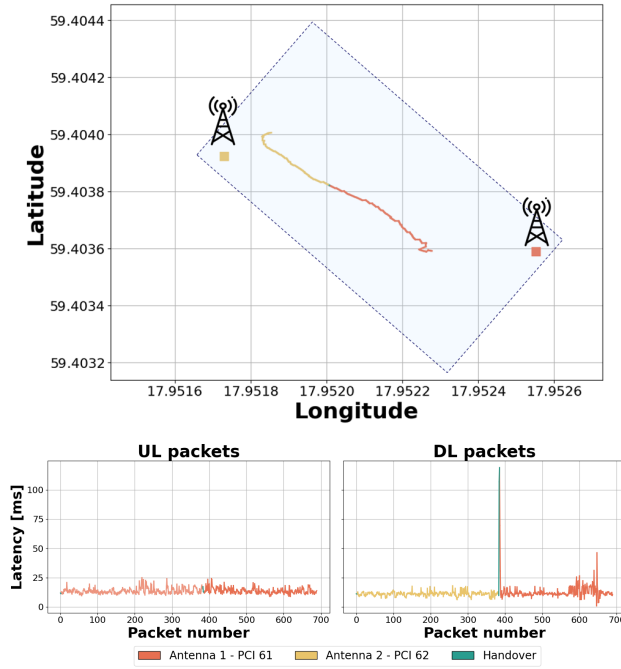
Fig. 6. Vehicle traveling along the handover route. The vehicle starts near Antenna 2 (yellow) and drives towards Antenna 1 (orange). During this route, DL traffic (server-vehicle) is logged and shown in the respective color. UL traffic (sensor-server), which always goes through Antenna 1, is also shown before (light orange) and after (dark orange) handover. Packets affected by the handover are shown in green.

Table I, while no additional features were added to help lower the handover latency. Fig. 6 displays the GPS coordinates and packet latency following the handover route in Fig. 2. Specifically, this allows us to know where and when these handovers occur, along with the latency at these times. By following this route in a controlled environment, we can repeat handover events to evaluate their consistency and the 5G C-V2X application performance when these transitions occur.

V. DISCUSSION

The performed tests, inspired from shared situational awareness, show that the presented testbed implementation stands as a fast and effective development platform. Moreover, it indicates that small-scale testbeds are suitable for evaluating advanced 5G C-V2X applications. For example, from the test cases under nominal conditions, we are able to track individual messages and measure the latency when they are of varying sizes and frequencies. From the test cases under overload conditions, we are able to observe the testbed's ability to stress test an implemented 5G C-V2X application when there are additional users in the same network. Finally, from the test cases under mobility conditions, we are able to see the effects of handovers, providing us a controlled way to evaluate these events in relation to the 5G C-V2X application. For future work, we are implementing the full 5G C-V2X shared situational awareness application along with an intelligent intersection application, similar to [19], and will evaluate both of their

real-life performances. In parallel to this implementation, we will also continue to investigate and implement new features, such as features limiting handover interruption time, that are being specified in 3GPP to enhance the network's ability to support 5G C-V2X applications

ACKNOWLEDGMENT

REFERENCES

[1] I. Soto, M. Calderon, O. Amador, and M. Urueña, "A survey on road safety and traffic efficiency vehicular applications based on C-V2X technologies," *Vehicular Communications*, vol. 33, p. 100428, 2022.

[2] Z. Amjad, A. Sikora, B. Hilt, and J.-P. Lauffenburger, "Low Latency V2X Applications and Network Requirements: Performance Evaluation," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, pp. 220–225.

[3] R. Miucic, A. Sheikh, Z. Medenica, and R. Kunde, "V2X Applications Using Collaborative Perception," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pp. 1–6.

[4] V. Narri, A. Alanwar, J. Mårtensson, C. Norén, and K. H. Johansson, "Shared Situational Awareness with V2X Communication and Set-membership Estimation," no. arXiv:2302.05224, 2023. [Online]. Available: http://arxiv.org/abs/2302.05224

[5] European Telecommunications Standards Institute, "Intelligent Transport Systems (ITS); ITS-G5 Access layer specification for Intelligent Transport Systems operating in the 5∼GHz frequency band." [Online]. Available: https://www.etsi.org/

[6] 3GPP, "The 3rd generation partnership project." [Online]. Available: https://www.3gpp.org

[7] 5G Automative Association (5GAA), "C-V2X explained." [Online]. Available: https://5gaa.org/c-v2x-explained/

[8] M. N. Tahir, T. Sukuvaara, and M. Katz, "Vehicular Networking: ITS-G5 vs 5G Performance Evaluation using Road Weather Information," in *2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pp. 1–6.

[9] F. J. Jiang, M. Al-Janabi, T. Bolin, K. H. Johansson, and J. Mårtensson, "SVEA: an experimental testbed for evaluating V2X use-cases," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. Macau, China: IEEE, pp. 3484–3489.

[10] Stanford Artificial Intelligence Laboratory et al., "Robotic operating system." [Online]. Available: https://www.ros.org

[11] G. Kueppers, J.-P. Busch, L. Reiher, and L. Eckstein, "V2aix: A multi-modal real-world dataset of etsi its v2x messages in public road traffic," *arXiv preprint arXiv:2403.10221*, 2024.

[12] OASIS MQTT Technical Committee, "MQTT Version 3.1.1 Protocol Specification," OASIS, Tech. Rep., 2014, available: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html.

[13] Cloud Native Computing Foundation and Synadia, "NATS." [Online]. Available: https://nats.io

[14] ZettaScale Technology, "Zenoh." [Online]. Available: https://zenoh.io

[15] J. Martin, J. Burbank, W. Kasch, and P. D. L. Mills, "Network Time Protocol Version 4: Protocol and Algorithms Specification," RFC 5905, Jun. 2010. [Online]. Available: https://www.rfc-editor.org/info/rfc5905

[16] "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems," *IEEE Std 1588-2019 (Revision of IEEE Std 1588-2008)*, pp. 1–499, 2020.

[17] ETSI, *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective Perception Service (CPS); Release 2*, etsi tr 103 562 v2.1.1 (2019-12) ed. The European Telecommunications Standards Institute, 2019. [Online]. Available: https://www.etsi.org/deliver/etsi_tr/103500_103599/103562/02.01.01_60/tr_103562v020101p.pdf

[18] L. ESnet, "iperf3." [Online]. Available: http://software.es.net/iperf/

[19] K. M. Arfvidsson, F. J. Jiang, K. H. Johansson, and J. Mårtensson, "Ensuring Safety at Intelligent Intersections: Temporal Logic Meets Reachability Analysis," in *2024 IEEE Intelligent Vehicles Symposium (IV)*.