Service performance measurement methods for 5G Connected and Automated Mobility use cases

Panagiotis Kontopoulos, Sokratis Barmpounakis, Maciej Muehleisen, Frederic Gardes, Roshan Sedar, Francisco Vazquez-Gallego, Ramon Casellas, Francesca Moscatelli, Xavier Vilajosana, Andreas Schimpe, Jérôme Härri and Soumya Kanti Datta

***Abstract*—The fifth generation (5G) network has provided opportunities for novel over the network applications and services, all with diverse demands for uplink and/or downlink latency, throughput and reliability. One sector, which poses strict requirements with regards to the aforementioned Key Performance Indicators (KPIs) is the Automotive -and more specifically the Connected and Automated Mobility (CAM), with a plethora of use cases and scenarios. Although numerous methods have been applied in the past, in order to properly measure the different aspects of the 5G networks' performance, as well as the domain-specific applications' KPIs, there is a need to shed more light on novel and efficient means to assess the impact of 5G deployment's performance on CAM use cases and scenarios. In this paper, a thorough methodology is presented for three key CAM use cases, namely Tele-operated Drivng, High-definiton Mapping and Anticipated Cooperative Collision Avoidance, Besides the methodology, measurement tools and selected KPIs, results from the large-scale trials' execution in European corridors are also presented and discussed, for each one of those use cases.**

***Index Terms*—Vehicles, 5G, Measurements, KPIs, Tools.**

I. Introduction

DURING the last years, in the context of the 5th Generation (5G) mobile communications system design and development activities, a plethora of solutions targeting diverse radio, network and management aspects have contributed in the radical evolution of the new system, demonstrating substantial gains comparing to the previous generation. Nevertheless, standalone or disconnected solutions may often omit to take into consideration several complexities and/or third-party bottlenecks of the end-to-end system operation, leading thus to non-realistic results and/or assumptions. Additionally, measurements of specific aspects of the 5G system's performance may differ considerably from previous (e.g., 4G or 3G) methodology [1].

Recently, the 5G Public Private Partnership (5G PPP)[2] released a white paper on the Service performance measurement methods over 5G experimental networks [3], which makes a thorough analysis on vertical use cases of various domains for their application-specific performance Key Performance Indicators (KPIs) and their mapping to the respective 5G system-specific KPIs. Its primary scope is to identify (based on architectural elements analysis, information flow, etc.) the potential impact on the service performance and user perceived quality. As this white paper highlights, the main challenge is to understand the relative influence of 5G network performance indicators to the vertical services. The KPIs mapping methodology includes three steps, i.e.: a) research on definitions and information derived from relevant 5G-PPP projects, standardisation bodies and respective alliances, such as ITU, NGMN etc., as well as definition of use cases from 5G-PPP projects' prespective; b) identification of relevant key service KPIs and their definitions that are of importance to the respective industry, and c) mapping of selected service KPIs on the respective network KPIs that impact the operation of the architectural elements that participate in the service provision process.

Automated driving and mobility is one of the key use cases, which have been identified for driving the 5G system innovation and potential, accompanied by numerous challenges and ultra-strict requirements [13], both for enabling the successful automated operations but also for ensuring that human safety will never be at stake. To this end, the Cooperative, Connected and Automated Mobility (CCAM) concept has been recently introduced; in this context, a plethora of diverse applications and scenarios have been described, such as Tele-operated Driving, Anticipated Collision Avoidance, Vehicle Platooning, etc. The realization of CCAM targets to radically increase the mobility safety by reducing accidents, improve road traffic efficiency, reduce environmental impact of road traffic, and foster new ways of creating revenue via reducing both capital and operational costs of mobility stakeholders.

Advanced validation trials have been initiated during the last years, presenting already results concerning both 5G end-to-end system (non-standalone or standalone) related KPIs, as well as vertical application-specific ones. Such trials focus on diverse vertical industries, such as the Automotive, Industry 4.0, Media and Entertainment, Energy, etc. and target to evaluate the end-to-end 5G system's performance, analytically measure, assess and analyze validation results and ultimately provide valuable insights with regard to the actual, realistic capacity of the new system.

A group of projects, funded by the 5G PPP Phase 3 [4] are among the ones, which already output results of such advanced validation trials. 5GCroCo [5] is such a 5G PPP Phase 3 project, which carries out CCAM trials in the cross-border corridor along France, Germany and Luxembourg, and which validates different 5G technologies of an end-to-end

P.K. and S.B. are with National and Kapodistrian University of Athens (NKUA). M.M is with ERICSSON GMBH (ERI-DE). F.G. is with ORANGE SA (Orange). R.S., F.V.G. and R.C. are with Center Tecnologic de Telecommunications de Catalunya (CTTC). F.M. is with NEXTWORKS (NXW). X.V. is with WORLDSENSING SL (WSE). A.S. is with Technical University of Munich (TUM). J.H. is with EURECOM (EUR). S.K.D. is with EURECOM (EUR)

deployment, such as Mobile Edge Computing (MEC), cross-Mobile Network Operator (MNO) handover, NFV MANO and SDN integrated architectures for cross-domain scenarios, end-to-end Quality of Service (QoS), predictive QoS with CCAM application adaptation, etc. Those technologies are validated in the context of three key use cases (UCs), namely Tele-operated Driving (ToD), High Definition Mapping (HD Mapping) and Anticipated Cooperative Collision Avoidance (ACCA).

In this work, we provide a comprehensive analysis on the performance evaluation methodology that has been followed in the context of the project, focusing on the three project use cases, as well as their specific requirements and KPIs; the tools that have been developed and utilized throughout the validation activities are also presented, along with indicative result samples that provide a deeper understanding on the analysis that was carried out.

II. Tests and Trials Presentation

In this section, an overview of the three 5GCroCo test cases are presented, along with their key requirements, trial site information and execution methodologies.

1) Tele-Operated Driving: Current automated driving vehicle prototypes prove the feasibility of truly driverless cars. Tele-operated Driving (ToD) can be leveraged as an enabling technology to smooth this transition, as edge cases remain which necessitate falling back on human operators. An overview of the ToD use case is shown in Figure 1. For ToD, an interface over the mobile 5G network is created that allows a human to remotely control a vehicle. Through such an interface, sensor and vehicle data, e.g., video feeds and current velocity, are transmitted from the vehicle to the vehicle control center (VCoC). There, the data are displayed to the human operator who generates control commands.

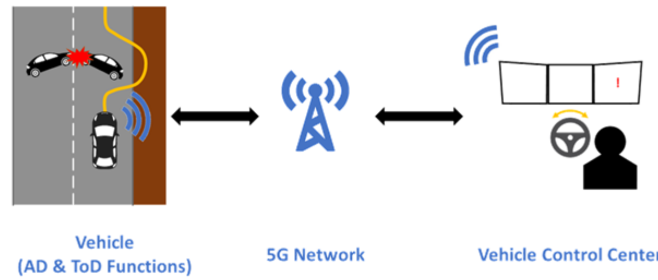

Fig. 1. Overview of Tele-operated Driving Use Case

ToD has very demanding requirements with respect to functional safety as errors generated by the automated vehicle system might cause injuries to passengers and other road users. Today's concepts for functional safety, like the one specified mainly in the ISO26262 standard [6], are not considering the case that vital parts of the system are developed not considering the rules specified in the ISO26262. To keep the possibility to provide functional safe ToD, concepts have to be developed that allow the existence of system elements that are not developed according to ISO26262 while still keeping functional safety under full control (this could be but is not limited to a safe control of the respective system elements). A functional safe and reliable control of the communication is one of the key needs. For this, 5G concepts like predictive Quality of Service (pQoS) are indispensable [14].

In addition, high quality of the information provided to the tele-operator, e.g., from cameras in the vehicle is necessary to allow safe tele-operation. This means that camera information together with data from other sensors must be provided to the tele-operator with short latency, high quality and high frame rates. This is only possible with new 5G uplink capacity, latency and reliability capabilities.

This test case is divided in four User Stories:

- User Story 1 – Remotely Controlled Maneuvering (direct control, low velocity) - The goal is to demonstrate how Tele-Operated Driving and 5G communication technology can be used to overcome scenarios in which an automated driving vehicle does not know how to handle an unexpected road blockage.
- User Story 2 – Remotely Controlled Path-based Driving (indirect control, low velocity) - Focus is to demonstrate how Tele-Operated Driving and 5G communication technology can be used to overcome scenarios in which an AD vehicle does not know how to tackle an unexpected road blockage. In this user story, the indirect control concept, which is less critical in terms of latency, is applied.
- User Story 3 – Remotely Supervised Control (high velocity) - The goals of the test cases for this user story are to evaluate the limits of ToD with 5G technology, analyze the handover and velocity limits and check cross-country interoperability.
- User Story 4 – Slim Uplink for ToD (indirect control, low velocity) - This user story demonstrates if a ToD session can be executed using a new approach. It implies to share periodically changing images instead of videos and to compensate this with additional vehicle perceived object information to the tele-operator. Furthermore, it has the goal to observe if this new approach allows the planning of future mechanisms that will be the key to calculate the overall resources needed across the vehicle and the VCoC. The main goal is to improve efficiency and scalability of the uplink data channel, while achieving comparable results as in previously described user stories, in which the main goal is to overcome scenarios where an AD vehicle does not know how to tackle an unexpected road blockage.

2) High Definition Mapping: The high definition map is one of the corner stones of an autonomous car. The basic functionality is to determine the vehicles position, which road and which lane is it in. Once the position has been established, information such as lane markings, lane connectivity and other information can be extracted from the map and used by the AD function. The HD map can also be used as the base upon which more dynamic information can be stored, e.g. road works and accidents.

Since the AD cars require the map data to be constantly up to date, the drives done by the map suppliers mapping vans are not sufficient. As many cars as possible need to be able to contribute to keeping the map up to date, using their connectivity to send sensor data to some back end, where the map can be updated if needed. Depending on the sensor type, this data can be quite heavy and require large bandwidth. In other cases, a low latency is required, so that changes can be made available to other cars as quickly as possible.

In this use case the 5G network and the MEC will be benchmarked using HD maps as the test subject and according to the aforementioned requirements.

Seamless availability of the autonomous and automated driving functionality is key for acceptance of such functionality. This is especially true in a fully autonomously driving car, where outages of the functionality would cause passengers not capable to drive themselves to be stuck. In order to get such seamless availability, it is very important to have accurate HD maps, especially the dynamic, fast changing parts, available any time and everywhere.

Such an order of magnitude of availability cannot be achieved with today's communication networks that are lacking full area coverage and a reliable prediction of the availability of connectivity. For HD maps a function like predictive Quality of Service that is currently discussed for 5G is essential to allow the aforementioned availability always and everywhere.

In future traffic scenarios there might be many autonomous vehicles active in quite small areas meaning that the HD map updates have to be provided for many vehicles at once. This leads to new requirements for high capacity or novel data distribution capabilities in downlink (e.g. efficient multicast) which are not fully available yet in 4G and below. Furthermore, the quality of the HD maps is highly correlated with the number of sensing vehicles that participate in the HD map generation. This means that, even though the amount of data for individual cars might be small enough to be covered by 4G Networks, the number of contributing communication units likely cause data traffic that goes beyond the capacity of today's networks.

HD map content hosting and receiving updates in MEC assures data is kept local where it is actually needed and enables rapid cycles of uploading detected HD map changes and distributing this change to vehicles in proximity. MEC hosting enables the MNO to have control over the full end-to-end path of the communication, which is not true when only relying on hosting in the public Internet.

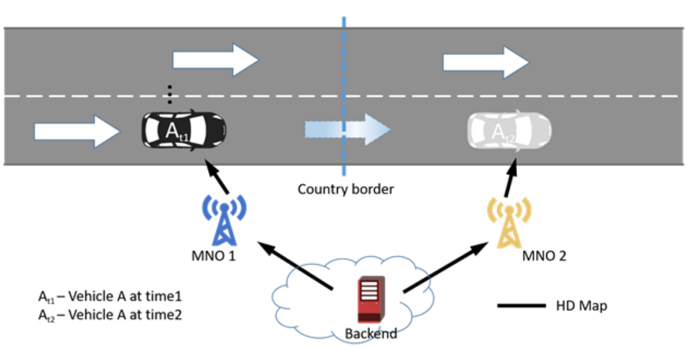

Fig. 2. HD Mapping User Story 1 – HD Map Downloading to the Car

The HD Mapping also is divided in four scenarios/test:

- User Story 1 – Streaming the Map to the Car - The goal of the test cases for this user story is to demonstrate how 5G communication technology can be used to enable automated driving by making sure that the car always has the latest version of the HD map available.
- User Story 2 – Uploading Map Deviations to the Cloud - Demonstrates how 5G communication technology can be used to enable automated driving by making sure that the map system in the cloud will catch changes in the environment as soon as possible, thus enabling the car to always have an HD map available which is as accurate as possible.
- User Story 3 – Closing the Loop - In this user story it is demostrated how 5G communication technology can be used to enable automated driving by making sure that the map system in the cloud will catch changes in the environment as soon as possible, update the HD map accordingly and then send the updated map to the car, thus enabling the car to always have an HD map available which is as accurate as possible.
- User Story 4 – Download Map Updates Supported by Network Performance/Cost Prediction (pQoS) - This user story is similar to user story 1. But before performing the download, the vehicle provides the route (waypoints) it intends to drive to the network and the estimated data volume to be downloaded. The network will provide a prediction of:

  Expected downlink throughput at the waypoints (Best Performance)

  Cost charged at the waypoints (Cost Reduction) With this information, the vehicle will select the appropriate location to perform the download.

3) Anticipated Cooperative Collision Avoidance: The Anticipated Cooperative Collision Avoidance (ACCA) application relates to the possibility to anticipate certain events in order to reduce the probability of collisions in situations when typical sensors partially will have no visibility or a short detection range (e.g. a few hundred meters). In certain situations, typical vehicle sensors (radars, cameras, lidars) will not have good enough performance to reliably detect and localize dangerous events on the road with sufficient level of anticipation. In such a situation, too late detection of a dangerous event will trigger a hard braking or a dangerous maneuver or even lead to a collision.

The ACCA service allows to anticipate the detection and localization of such dangerous events to induce smoother and more homogeneous vehicle reaction. It is based upon the use of off-board servers that collect and process the available information. V2N2V type of communication is prioritized but direct communication is not precluded. The vehicles will upload a set of information such as their warning status (position, speed, heading etc.) on themselves (e.g. encoded in CAM's) or on remote elements like the road side infrastructures using their local perception based on-board sensors (e.g. encoded in CPM's) [15]. At the same time, they will share a set of detected events (via the use of hazard reports, e.g. encoded in DENMs). This data is uploaded into specific off-board servers. These data will be used by Mobile Edge and Public Internet deployed servers to create an off-board Distributed Dynamic Map (DDM) which handles and consolidates all collected information based on a known road topology. The ACCA application can be considered as a first step towards an off-board eHorizon reconstruction.

5G with MEC is required for ACCA use cases for the following reasons:

- Need of high and guaranteed reliability of the connectivity between the vehicle and the off-board, MEC-hosted Geoservice [16],
- Need of low and guaranteed latency of the connectivity between the vehicle and the off-board Edge MEC-hosted Geoservice,
- Need of backend interconnection between the central Traffic Management System hosted on the Public Internet and the MEC-hosted Geoservice. It shall be noted that the ACCA use cases will not stress the 5G capability of high throughput per vehicle. To provide useful information, the MEC service shall demonstrate that it can:
- provide trustable information by running fusion of hazard information coming from multiple sources (vehicles and backends),
- provide information with a low latency,
- aggregate information from multiple sources (vehicles and backends) and of different nature, to deliver contextual information, which can be more easily used by ADAS systems.

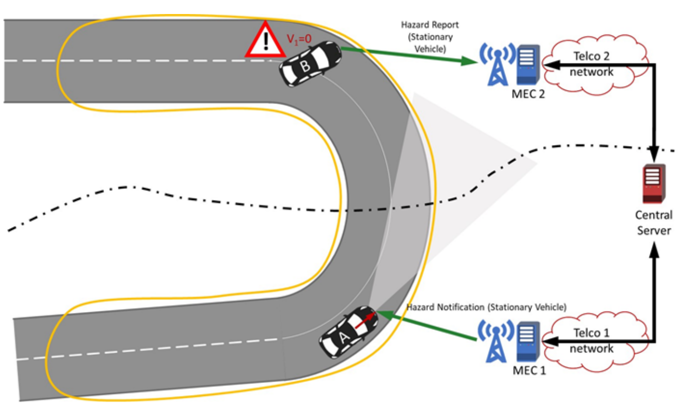

Fig. 3. ACCA User Story 1 - Ego Detected Hazard (Cross-country)

In this Use Case there are three scenarios:

- User Story 1 – Ego-detected Event - Two cases are considered for this user story:

  1) Vehicle B had an accident and is assumed to be on the same lane as the vehicle A. It sent a hazard report (e.g. ego DENM) to the local MEC host. Vehicle A subscribes with the Geoservice to be notified about hazard in its ROI (Region Of Interest). When its ROI covers the hazard, Vehicle B is informed by the Geoservice with an immediate response following its subscription

  2) Vehicle A has subscribed with a large ROI. While the ROI is the valid ROI for the vehicle A, Vehicle B has an accident. It sends a hazard report (e.g. ego-DENM) to the Geoservice which notifies Vehicle A.
- User Story 2 – Stationary Vehicle Remotely Detected Event - In this story the focus is on the detection of hazard on the road with the use of the MEC infrastructure. In two occasions, with the use of the vehicle ego information or the vehicle on-board sensor detection, the Stationary Vehicle hazard is computed and then shared by the Geoservice, depending on patterns on the shared values.
- User story 3 – Traffic Jam Remotely Detected Event - This user story is similar with User Story 2 but now the event that the Geoservice triggers is a Traffic Condition with many vehicles instead of a Stationary Vehicle concerning a unique mobile object.

A. KPI Description

In the following section the overview of the KPIs will be presented that were used in the use cases above. Each use case has a specific set of KPIs used to validate the tests and trials.

1) Tele-Operated Driving: Regarding the ToD, it was nessecary to validate not only the KPIs from the network's perspective, but also from the car's side in order to have the a complete overview of what is expected from such scenarios.

- KPI #1.1 – Manoeuvring Range: This is the distance covered while the vehicle is being tele-operated. The values are being calculated and measured in the vehicle.
- KPI #1.2 – Service Range: This is the distance between the VCoC, i.e., the distance over which data is transmitted. The values are being calculated in post-processing.
- KPI #1.3 - Information Exchanged and Estimated Payload: This is the data volume and data rate exchanged during the execution of the user story. The values are being calculated and measured by the receiving side, based on packet timestamps and sizes. Results will be given for both, uplink and downlink.
- KPI #1.4 – Application Level Latency: This is the time it takes for a packet to be transmitted by the mobile 5G network. It is calculated and measured by the receiving side, based on packet timestamps and does not include the actuator delay, the sensor delay nor the in-vehicle network delay. This presupposes that the sender and receiver have synchronized clocks. Results will be given for both, uplink and downlink.
- KPI #1.5 – Application Level Reliability: This is the complementary packet error rate, i.e., 1 minus the number of lost packets per number of transmitted packets. The values are being calculated and measured by the receiving side, based on packet sequence numbers. Results will be given for both, uplink and downlink.
- KPI #1.6 – Age of Information: This is the time it takes for an event, happening in front of the vehicle's camera sensors, to be displayed on the screens in the VCoC. It is only measured in the uplink direction. As compared to the uplink Application Level Latency, the measurements incorporate the latencies in the complete application, e.g., the camera sensor, the processing algorithms and the screens.
- KPI #1.7 – Positioning Accuracy: This specifies how precise the global position, obtained via a GPS / Inertial

Measurement Unit (IMU) or odometry is measured. This is relevant for performing path tracking control.

- KPI #1.8 – Deviation from Desired Path: This is the extent to which the actually driven path deviates from the desired path, caused by tracking errors of the vehicle-intern path tracking controller. The lateral deviation is the distance from the vehicle's position to the desired path (length of the perpendicular). The orientation deviation: Difference between the desired orientation and the actual orientation of the vehicle.

2) High Definition Mapping:

- KPI #2.1 - Application Level Reliability: It is the percentage of successful service invocation, calculated as $P_{success} = 1 - \frac{n_{failed}}{n_{total}}$ where n_{total} is the number of map tiles or detected map deviations that are requested to download or upload, and n_{failed} is the number of fails in which the failure is assumed if download or upload does not complete within maximum time period. A download or upload will not finish within maximum time period if the corresponding downlink or uplink throughput, respectively, is too low.
- KPI #2.2 - Breakdown (Sub-KPI) Download / Upload Throughput: For HD Maps, information is exchanged over the mobile network from the vehicle to the HD Map backend/server and vice-versa. In one direction, map data is sent from map server to the vehicle, in other direction, detected map deviations is sent from vehicle to the map server. The download/ upload throughput presents the measurements of information payload normalized with the transmission time duration.
- KPI #2.3 - Age of Information: The Age of Information, needs to facilitate an appropriate reaction time of HD Map changes to the current situation. It is the time duration from the lead vehicle starting to send map deviations to the map server, where the corresponding map tile is updated and a new download is triggered to the following vehicle, until the following vehicle finishing the download of the updated map tile.

3) Anticipated Cooperative Collision Avoidance:

- KPI #3.1 - Application Level Latency: a) User Story 1: Time when the Hazard Notification is received by Vehicle A minus time when the event is detected by the stationary Vehicle B. b) User Story 2 and 3: Time when the Hazard Notification (stationary vehicle / traffic jam warning) is received by Vehicle A minus time when the relevant message (CAM, other DENM, CPM or whatever other message is used to detect the event) is generated by the stationary vehicle (User Story 2) or vehicles in the traffic jam (User Story 3).
- KPI #3.2 - Application Level Reliability: Number of received Hazard Notifications by Vehicle A divided by number of new or updated Hazard Reports generated by stationary Vehicle B, provided the Hazard Notification is received within 1s. A notification of an event which is older than 1s is considered as obsolete and discarded. The values are post-processed based on measurements in vehicle A and B. For more than one receiving vehicle it is measured per receiving vehicle. For User Story 2 and 3 also the ratio of successfully received messages (e.g. CAMs) used to detect the event in the backend is evaluated.
- KPI #3.3 - Service Provisioning Time: The duration required for setting up the end-to-end logical network service and configuring the related network and computing resources that are required for guaranteeing the proper operation of the virtualized service. In particular, in a cross-border scenario, the measured Service Provisioning Time represents the overall time required for provisioning a cross-border Network Slice and the corresponding logical network services, then including: - The time consumed at the centralized Service Orchestrator [3] for coordinating the cross-border Network Slice provisioning. - The time consumed at each vMNO's domain for setting up the Network Slice Subnet of its competence, configuring the network/computing resources and running the Virtual Network Functions composing the network service. - The time consumed at the Service Orchestrator for checking the Network Slice Subnets status and verifying the result of the end-to-end provisioning procedure.

B. Methodology

For determining the confidence of the results a typically applied method to assess result confidence is to calculate Student-t confidence intervals from independent experiments [7]. In this context of CCAM trials it would mean to repeat the same experiment several times keeping parameters, which are under control of the executing persons, identical every time while the random influences will be different each time but should follow the same statistics. The experiments and results obtained would then be independent and identically distributed (IID). For each experiment the KPI of interest could be determined, e.g. the Application Level Latency or Reliability, and the average value over all experiments can be calculated together with its confidence.

Alternatively, a single experiment, that should be very long, can be executed and then the results (e.g. Application Level Latency for each received message) are separated into batches where each batch is considered an independent experiment. Then the KPI for each batch is calculated along with the average over all batches together with the confidence interval. This method is known as Batch-Means [8]

The methods above require the number of experiments and/or batches to be increased and/or to increase the runtime of the experiment(s) to improve result confidence. Within each experiment, each received measurement sample (e.g. message latency) contributes to improve the confidence. It is therefore of interest to find a method to calculate result confidence and its improvement based on each received measurement sample. The question if a latency is above or below a certain threshold is a Binomial Experiment where a set of samples can have two possible outcomes, namely "below or equal" or "above threshold". In case of consecutive samples being independent, the formula for Binomial Proportion Confidence Intervals [9] can be used: $\hat{p}^{+}_{-} z\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$ with z = 1.96 for the

95% confidence interval, n is the total number of samples and $\hat{p}$ the fraction of samples with desired outcome, e.g. “below threshold”. The term behind the plus-minus corresponds to the symmetric confidence interval size around $\hat{p}$.

Independent consecutive measurements, especially when it comes to latencies, are rarely encountered in reality. Latencies are correlated as they e.g. originate from buffers that do not suddenly change their fill level or from retransmissions during bad network conditions that usually also last for a while and do not (dis)appear from one message transfer to the other. Still, the formula can be used for a worst-case estimation of required samples m for a certain confidence.

The Limited Relative Error (LRE) Algorithm was designed to overcome the limitation of requiring consecutive samples to be independent [10]. The mathematical foundation of the algorithm is described in [11] along with its practical implementation in [10].

The algorithm was particularly designed for network latency measurement experiments. The origin of the term Limited Error comes from the fact that in one step the ”randomness” of the system itself is assessed by estimating its standard deviation. Traditional confidence intervals can be large for two reasons: either the experiment was not repeated enough times, or it contains a lot of ”randomness”. By normalizing to the estimated standard deviation, two experiments, that only differ in their standard deviation, achieve same confidence interval sizes with same number of repetitions. Figure 4 illustrates

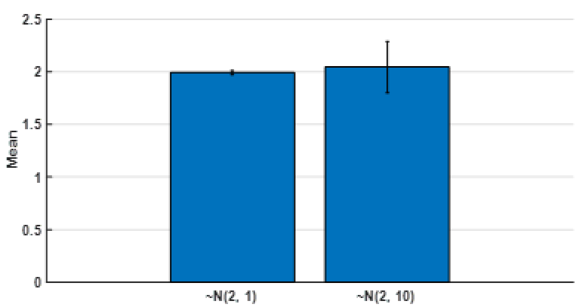

Fig. 4. Mean Value of 10000 Normally Distributed Random Samples with Mean Value 2 and Standard Deviation 1 and 10; Confidence Interval for 10 Batches with 1000 Samples per Batch

the influence of the “randomness” on Student-t confidence intervals. 10000 random samples were drawn with same mean value 2 and standard deviation 1 and 10 for the left and right result bar, respectively. The samples were split into 10 batches with 1000 samples in each batch, according to the Batch-Means method, and the 95% confidence interval was calculated. For the lower standard deviation, the confidence interval is only around 1% from the estimated mean value, for the higher standard deviation it is 10 %. In both cases the true mean value 2 is within confidence but for standard deviation 10 also 2.2. is within 95% confidence. The LRE Algorithm determines an error that is independent of the “randomness” of the experiment.

The algorithm extends the model of a Binomial Experiment to a two state Markov-Chain where besides the probability of the outcome also the transition probability from one state to the other, or to stay within the same state, is considered. The algorithm therefore estimates state probabilities, which e.g. correspond to probabilities of exceeding a latency limit x, and transition probabilities, being the likelihood to exceed the limit again or to obtain a measurement that is below the threshold.

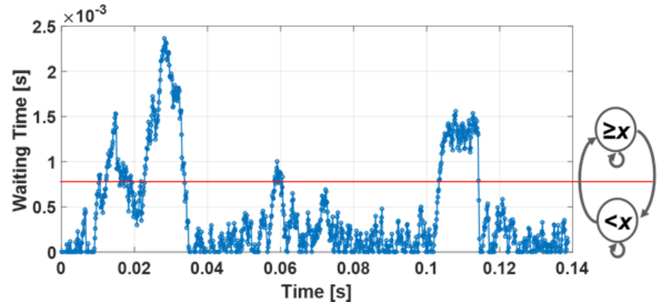

Fig. 5. Mapping of Latency Result Samples to a Two-state Markov-Chain Determining if the Result is Above or Equal to x = 0.75 ms or below.

Figure 5 illustrates the two-state Markov-Chain used for the LRE Algorithm when set to determine the relative error of an experiment that should determine the probability of a latency to exceed 0.75 ms. While Binomial Proportion Confidence Intervals would only consider the ratio of samples $\hat{p}$ encountered in a state and the total number of samples n, the LRE Algorithm also considers the state transition probability and this way also considers correlations of consecutive samples. It is therefore important that results are fed into the algorithm in the same order they were collected.

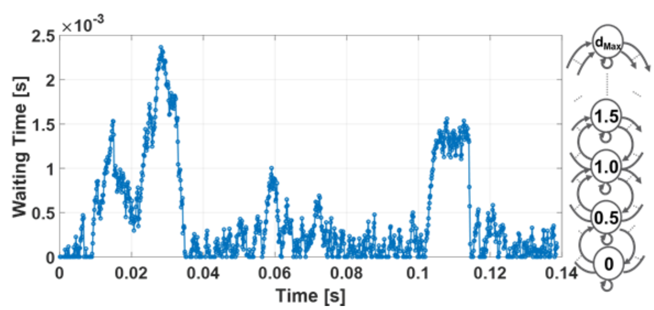

Fig. 6. Actual LRE Algorithm Implementation with Several “Buckets” between a Minimum (0) and a Maximum (dMax) Value

Figure 6 shows the actual implementation of the algorithm that does not require to define a certain threshold but is set to a maximum and minimum value and the interval (bucket) size for each state. The algorithm collects the number of “visits” to each state, which corresponds to the histogram. It furthermore saves the “transitions” for each state. These are defined as events of passing through the state. For example, if a value between 1.5 ms and 2 ms is measured, corresponding to state “1.5” and then one between 0 ms and 0.5 ms, corresponding to state “0”, the transition counter for state “1.0”, “0.5”, and “0.0” will be increased along with the visit counter for “0”. The relative error for a state can only be calculated if a minimum number of visits and transitions was recorded. As a rule of thumb, it makes no sense to use the algorithm with less than

1000 samples. The following conditions must be fulfilled for LRE to be able to calculate the relative error, with n the total number of samples, v(i) the number of visits to state i and t(i) the number of transitions for state i:

$n \geq 1000$

$v(i) \geq 100$, $n – v(i) \geq 100$ (collect min 100 samples left and right of state I; not enforced for max. and min. state)

$t(i) \geq 10$ (require min. 10 transitions)

$v(i) – t(i) \geq 10$ (have at least 10 more visits than transitions towards higher states)

$(1 – v(i)) – c(i) \geq 10$(same as before towards lower states)

So, a result is considered not confident either because the relative error is exceeded, which is typically set to 5%, corresponding to 95% common confidence interval value, or because there were not enough visits and/or transitions to the state to calculate the relative error. The interval (bucket) size can influence the number of visits and transitions but experience shows that adjusting it usually has minor influence on the point where the LRE algorithm determines the 5% error to be exceeded and the point from where on the relative error cannot be calculated anymore.

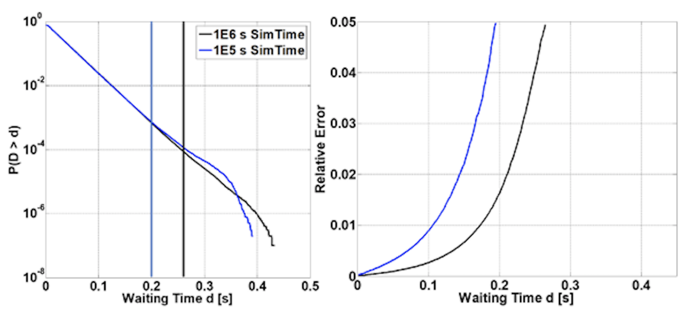

Fig. 7. CCDF (Left) of Queue Waiting Times from a Simple Network Simulator and Relative Error (Right) Determined by the LRE Algorithm for 0.1 and 1 Million Seconds Simulation Time

The left side of Figure 7 shows the CCDF of the delay distribution of the queueing delay in a generated with a simple network simulator . The result can be mathematically calculated and is exponentially distributed and therefore linear in the logarithmic representation used in the figure. For very large delays the shape obviously diverges from being linear. On the right the figure shows that the LRE Algorithm determines the error to be above% already from 0.2 s for 0.1 million second simulation time and 0.27 s for 1 million seconds. Looking at the corresponding y-axis values, the 99.9-percentile and 99.99-percentile were determined confidently this way. In this case a tenfold increase of samples also allowed a tenfold increase in determined percentile.

C. Tools

1) KPI Measurement Platform: Although each use case has its own validation method and its own KPI list, a common tool is used: we provide a performance workbench platform for each component involved into a use case. The purpose of this evaluation platform is firstly to monitor the health of the overall system in real time, by verifying that the logs show the expected performance.

On the one hand, we must ensure that the demonstration respects a certain level of quality. On the other hand, it is simpler to handle potential bugs and to monitor the complete system by collecting the logs. For example, we're able to quantify the elapsed time into each component for each message (in milliseconds). So, we can manage to handle both of these requirements deploying a complete stack to search, analyse and visualise data.

The monitoring platform collects measurements from each component deployed into the 5GCroCo project [5], ensuring that the end-to-end system works properly in real time and polling data to compute figures and to produce visualizations for the final report.

The goal is to provide a complete solution, including the server and the client parts. We choose the Elastic Stack solution, with Elasticsearch for storage and search, Logstash for data ingestion and transformation and Kibana for visualization. This allows us to compute indicators on the runtime with a specific format defined into the document.

Each component sends logs (partially or fully) to centralize, save, share and merge them with the others components. The monitoring platform aggregates many aims at the same time. • Collect from the Mobile edge computers and the public internet server • Ensure a low effort on implementers • Run offline to avoid network interference • Run online and require minimal remote supervision • Be flexible enough to handle different measurements As seen in Figure 8, each component develops its own client to share data. Every component in the MEC should send the required logs to a Logstash server installed in the same MEC in order to avoid having one Logstash instance handling all the MEC logs. If the component is packaged as a Docker container, the Gelf log driver can be used to automatically redirect the standard output device (stdout) to a given Logstash. The same collection method can be used for components in a public cloud with a dedicated Logstash in every cloud. The logs are then stored in an Elasticsearch instance hosted on a public cloud so that it can be accessible to all Logstash instances and all project partners from anywhere. Another solution may be to host an Elasticsearch instance in every MEC and synchronize their data with the cloud Elasticsearch: this would allow us to batch the new data every few seconds and avoid continuous connection but can be technically challenging.

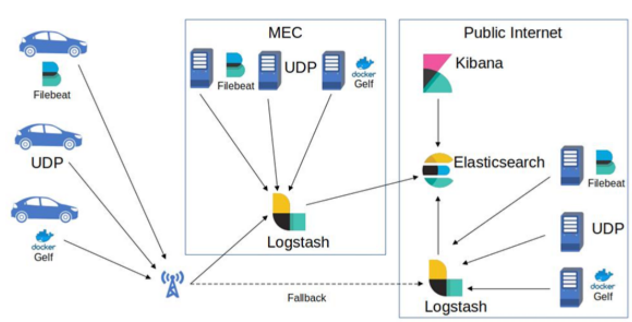

Fig. 8. Each Component Sends Its Message to the ELK System.

2) Arduino module for End-to-end Latency measurement: This module is used to measure Age of Information of the ToD application in the VCoC. The delay measurement principle, proposed in [17], is illustrated in Figure 9. The tool measures the period of time it takes for an event (blink of an LED) happening in front of the camera sensor to be detected by a photo transistor (PT) on the display.

Both, the light source and light sink, are connected to an Arduino single-board micro controller, which performs the delay measurements and suppresses noise. These end-to-end latency measurements, including the delays of the camera, the video encoding and decoding algorithms, the through a wired connection and finally the display, can then be collected by a PC via a USB connection.

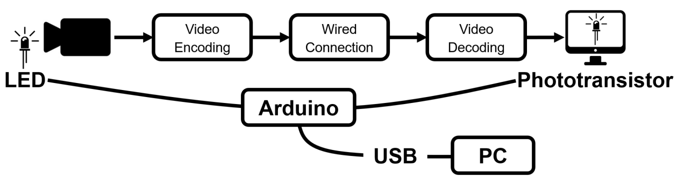

Fig. 9. Delay Measurement for End-to-End (Glass-to-Glass) Latency [17]

3) Volvo KPI Measurement Platform: For the HD Mapping use case also the KPI measurement platform is used to enable a real-time dashboard for demonstrations. In addition to that the HD Mapping use case reports measurements to another KPI measurement system which is also based on Elastic Stack. For the first round of test and trials this assured the required flexibility with changes to the measurement system sometimes even done between two trials after on-site analyses revealed the need to adjust something, e.g. adding further context information to measurements.

4) Orange HMI: The Orange HMI is a dedicated V2X Android application developed in-house by Orange. It is used in the ACCA use case for two different purposes: Real-time Remote Monitoring HMI On-board HMI

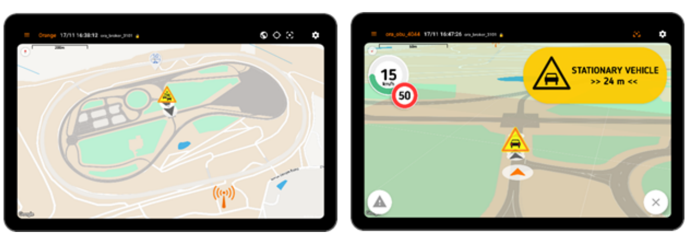

Fig. 10. Orange HMI in Monitoring Mode

5) ERI Network Delay Measurement PC: The car PC is hosting the backend side of the experiment. The network virtualization technique "network namespaces" is used to prevent data to be delivered locally through IP routing and instead to be sent over the radio network. On the backend side an MQTT Broker is deployed, such as done for the Geoservice. A corresponding sending client (MQTT Publisher) and receiving client (MQTT Subscriber) are also deployed allowing to obtain latency results closely resembling what is expected for small- and large-scale tests and trials, especially with regard to radio network.

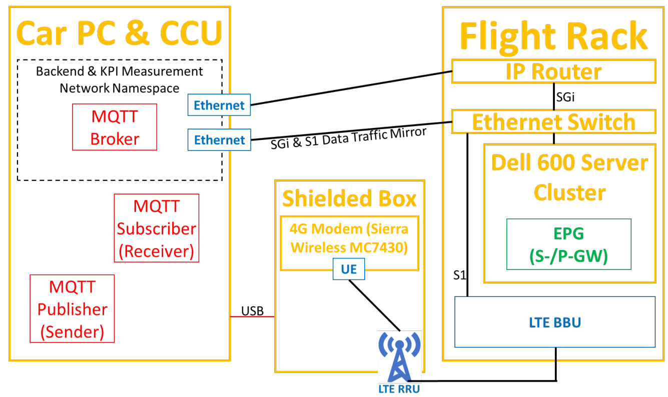

Fig. 11. End-to-end and One-way Latency Measurement System

D. Result Samples

In this section, the results from different scenarios are presented, captured with the tools described above.

1) Use Case 1 - ToD: The uplink latencies of signals transmitted during trials at the German-Luxembourgish border (D-L) are plotted over time in Figure 12. The latencies of two different signals are shown. The packets were transmitted via MQTT (TCP) from the vehicle to the VCoC. The connection was established from the vehicle in Luxembourg, connected to a 5G network, to the VCoC in Saarbruecken (Germany) through a VPN tunnel, which added a few milliseconds. However, it can be observed that the measured latencies are consistent. The average latency measured over all messages and signals was 17.9 milliseconds.

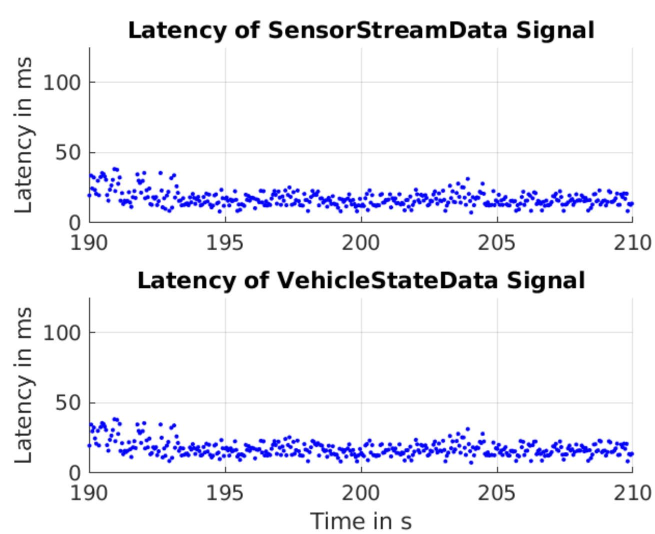

Fig. 12. Uplink Latencies during D-L Trials

Another result is the measurement of the age of information, using the measurement tool described in C paragraph 2. For 100 samples, the histogram in Figure 13 shows the frequency (number of samples times) over the time in milliseconds it took for the LED blink in front of the camera sensor to be detected by the phototransistor in front of the VCoC display

(G2G Latency). This measurement does not include the mobile network. Thus, only the latencies in the hardware and software components of application are measured. The average G2G latency measured as approximately 130 milliseconds.

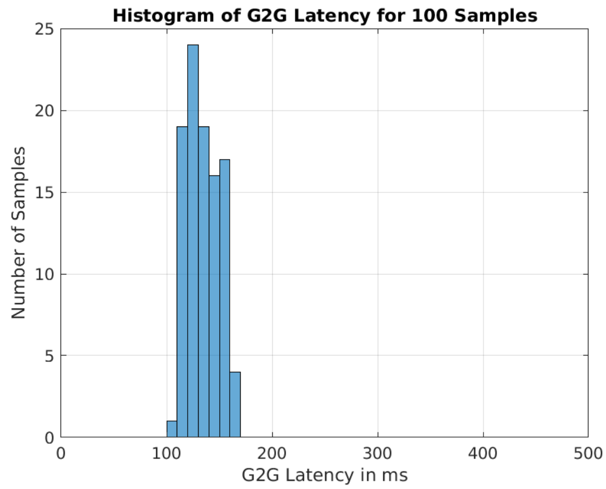

Fig. 13. Age of Information Measurement

2) Use Case 2 - HD Mapping: Figure 14 illustrates the influence of the different abovementioned effects on the instantaneous TCP throughput. It is shown for 4G, 5G as well as for public Internet and MEC. A tile transmission is finished when the integral over the respective instantaneous throughput equals the tile size in Byte. The difference between 4G and 5G is in the maximum throughput and the RTT making it difficult to precisely determine which of them has which effect on the overall result. Furthermore, different frequencies used, with 5G always being higher than 4G, result in changes of the maximum throughput due to changing radio channel conditions differ between the two.

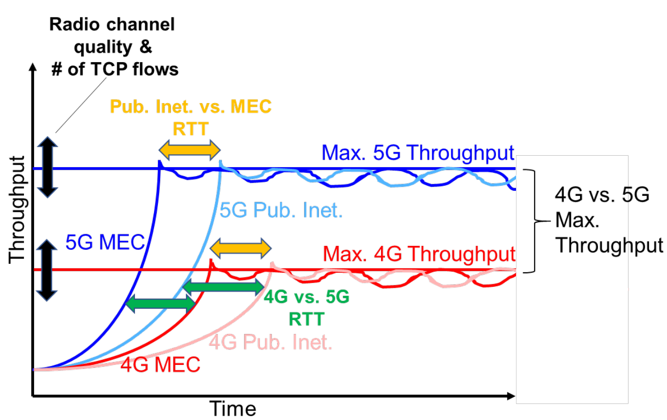

Fig. 14. Illustration of Different Influences on TCP Throughput

Figure 15 shows the average over all mean tile download throughputs for 5G and 4G with MEC and public Internet hosted HD Map Server.

Mean values in Figure 16 and corresponding CDF in Figure 17 show that MEC hosting provides slightly higher throughput than public Internet hosting. It is improved from 10.7 Mbit/s

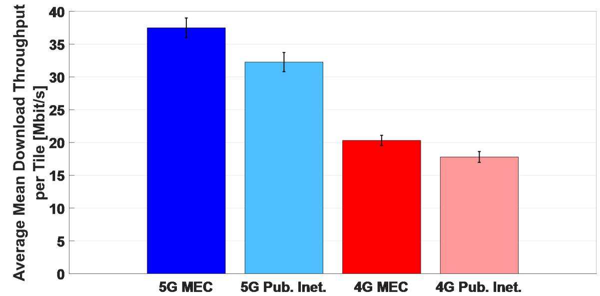

Fig. 15. Average over all Mean Tile Download Throughputs for 5G and 4G with MEC and Public Internet Hosted HD Map Server

to 12.9 Mbit/s (+20.6 %) and from 38.6 Mbit/s to 44.9 Mbit/s (+16.3 %) for 4G and 5G, respectively.

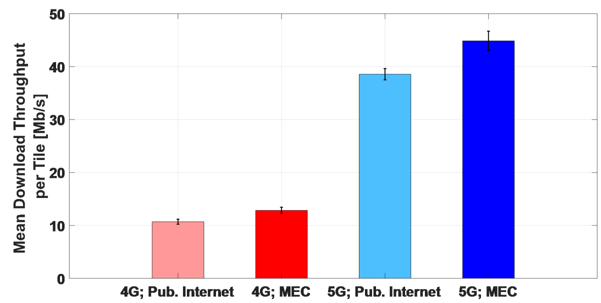

Fig. 16. Mean Download Throughput per Tile with 95 % Confidence Intervals

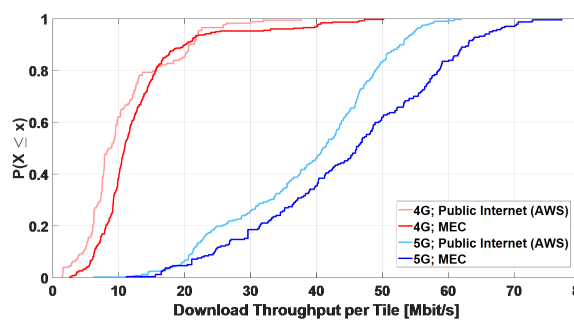

Fig. 17. CDF of Download Throughput per Tile

3) Use Case 3 - ACCA: Figure 18 visualize the mean values and 95 % CIs for MEC and public Internet hosting. The PSA2-5G (Peugeot S.A.vehicle with 5G equipment used in User Story 2) result of 23.2 ms mean Application Level Latency corresponds to the expectation. 13 ms RTT were measured with 700 Byte large Ping packets meaning that approximately 10 ms originate from application processing delays in the client and backend. For PSA1-5G (Peugeot S.A.vehicle with 5G equipment used in User Story 1) a similar result was expected but the obtained 48.6 ms are much higher. As the same backend was used, the origin must be in the application client, the Car PC hosting it or the CCU. This will be further investigated. The result for PSA-4G (Peugeot S.A.vehicle with 4G equipment) is 18 ms higher than for PSA2-5G (Peugeot

S.A.vehicle with 5G equipment) and corresponds to the expected latency difference between 4G and 5G. For RSA-5G (Renault S.A.vehicle with 5G equipment) the achieved mean Application Level Latency of 18 ms is slightly lower than for PSA2-5G resulting from different application processing delays on the client and the backend. Measurements for public Internet hosted Geoservice were only conducted for PSA2-5G and PSA-4G as they equally apply for all other vehicles.

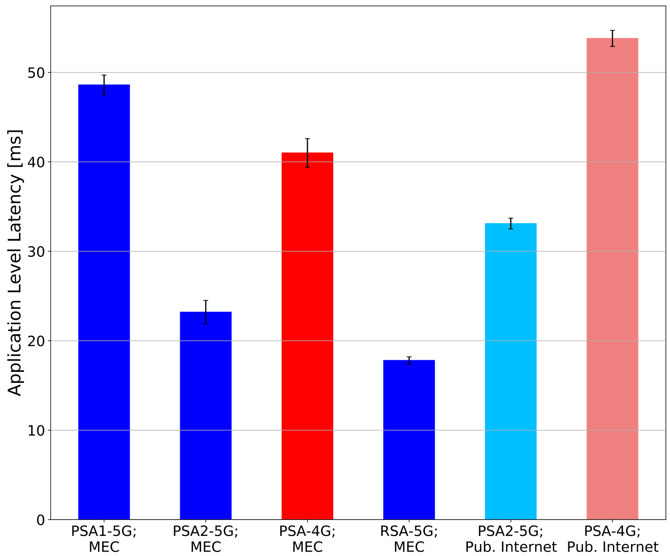

Fig. 18. Mean of ACCA Application Level Latency for MEC and Public Internet Hosted Geoservice with 95 % CIs

The CCDF of the Application Level Latency in Figure 20 shows that 90 % of the measurements from the PSA2-5G and RSA-5G vehicles are below 25 ms. This corresponds to the Ping RTT being 19 ms for 90 % of the samples assuming around 6 ms to 10 ms are added by application processing delay in the client and backend application. The 90 % is shifted by around 25 ms for the PSA-4G vehicle compared to PSA2-5G. This corresponds to the delay difference between the 4G and 5G system. For the other 10 % of the measurement occasional spikes in the application processing delay in the client and/or the backend must be assumed. They can be caused by occasional high loads on the CPUs of the hosts and Car PCs.

The CCDFs in Figure 20 show for at least 75 % of the measurements the expected behaviour of similar latency distribution with constant shifts between 4G and 5G of around 13 ms and 10 ms between MEC and public Internet hosted Geoservice.

III. Conclusion

In this work, we provided a detailed analysis on the approach and methodology that was followed in one of the pioneering EU 5G-PPP Phase 3 projects for Connected and Automated Mobility, namely 5GCroCo. Three key use cases have been thoroughly presented, namely Tele-operated Driving, High-definiton Mapping and Anticipated Cooperative Collision Avoidance, along with their evaluation KPIs. The tools that were deployed for acquiring the various measurements,

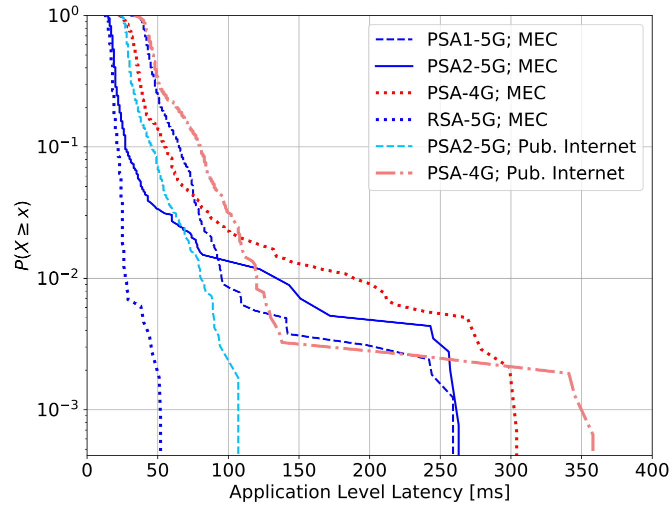

Fig. 19. Mean of ACCA Application Level Latency for MEC and Public Internet Hosted Geoservice with 95 % CIs

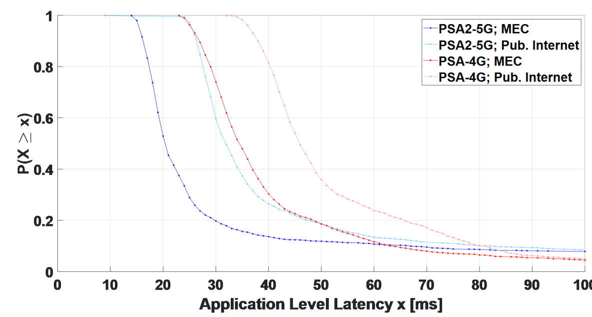

Fig. 20. CCDF of Application Level Latency

towards analysis and validation have been also presented. Last but not least, indicative results are presented towards providing a deeper understanding on the analysis that was carried out. This work overall connects a number of methodological and theoretical approaches for 5G CAM trial validation, with real world findings, which can provide valuable insights for the numerous forthcoming 5G and beyond-based trials to follow.

Acknowledgment

This work is part of the 5GCroCo project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 825050.

References

[1] Zoran Lazarević et all, *Benchmark measurements in 5G networks*, white paper, www.ericsson.com/4a0aa7/assets/content/22645e8e919f4fd38c4b28dd8daa661c/measure in-5g-networks.pdf ERICSSON, Germany, 2020.

[2] 5G Infrastructure Public Private Partnership (5G PPP), url: https://5g-ppp.eu/, url: https://5g-ppp.eu/ [accessed 7/2021]

[3] Markopoulos, Ioannis, Valcarenghi, Luca, Mesogiti, Ioanna, Taferner, Manfred, Boleguin, Patrice, Bouali, Faouzi, … Trichias, Konstantinos., *Service performance measurement methods over 5G experimental networks*, ICT-19 performance KPIs http://doi.org/10.5281/zenodo.4748385, 2021.

[4] 5G Infrastructure Public Private Partnership (5G PPP) Phase 3, url: https://5g-ppp.eu/5g-ppp-phase-3-projects/ [accessed 7/2021]

[5] H2020 5GCroCo EU project, url: https://5gcroco.eu/ [accessed 7/2021]
[6] ISO 26262-1:2018, Road vehicles — Functional safety — Part 1: Vocabulary, url: https://www.iso.org/standard/68383.html [accessed 7/2021]
[7] E. W.Weisstein, *Confidence Interval From MathWorld– A Wolfram Web Resource*, Online Available: https://mathworld.wolfram.com/ConfidenceInterval.html.
[8] P. Lassila, *Simulation of data networks -Collection of statistics and their analysis, part 2*, Online https://www.netlab.tkk.fi/opetus/s383148/s07/luennot/E_statanalysis2.pdf, 2007.
[9] S. A. Wallis, *Binomial confidence intervals and contingency tests: mathematical fundamentals and the evaluation of alternative methods*, Journal of Quantitative Linguistics 2013.
[10] C. G. a. F. Schreiber, *The restart/lre method for rare event simulation*, in Proceedings Winter Simulation Conference Coronado, USA: California, 1996.
[11] F. Schreiber, *Time efficient simulation: the LRE-algorithm for producing empirical distribution functions with limited relative error*, AEU. Archiv für Elektronik und Übertragungstechnik, vol. 38, p. 93 1984.
[12] H. Kopka and P. W. Daly, *Service performance measurement methods over 5G experimental networks*, 3rd ed. Harlow, England: Addison-Wesley, 2020.
[13] S. Euler, A. Pfadler, L. F. Ferreira and H. Zhao, *Spectrum Needs of Cooperative, Connected and Automated Mobility, 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), 2021, pp. 1-6, doi: 10.1109/VTC2021-Spring51267.2021.9448778.* Helsinki, Finland: 2021.
[14] Kousaridas, Apostolos and Manjunath, Ramya P. and Perdomo, José and Zhou, Chan and Zielinski, Ernst and Schmitz, Steffen and Pfadler, Andreas, *QoS Prediction for 5G Connected and Automated Driving, IEEE Communications Magazine, 2021, vol. 59, p. 58-66, num. 9, doi: 10.1109/MCOM.110.2100042* 2021.
[15] R. Sedar et al., *Standards-Compliant Multi-Protocol On-Board Unit for the Evaluation of Connected and Automated Mobility Services in Multi-Vendor Environments, Special Issue Convergence of Intelligent Sensing, Networking, and Computing Technologies for ITS, 2021, vol. 21, num. 6* London, UK: 2021.
[16] R. Vilalta et al., *Vehicular Message Exchange in Cross-border Scenarios Using Public Cloud Infrastructure, 2020 IEEE 3rd 5G World Forum (5GWF), 2020, pp. 251-256, doi: 10.1109/5GWF49715.2020.9221221* Bangalore, India: 2020.
[17] C. Bachhuber and E. Steinbach, *A system for high precision glass-to-glass delay measurements in video communication, 2016 IEEE International Conference on Image Processing (ICIP), 2016, pp. 2132-2136, doi: 10.1109/ICIP.2016.7532735* Phoenix, Arizona: 2016.