KQI Assessment of VR Services: A Case Study on 360-Video Over 4G and 5G

O. S. Peñaherrera-Pulla, *Member, IEEE*, Carlos Baena, Sergio Fortes, *Member, IEEE*, Eduardo Baena, and Raquel Barco

***Abstract*—Extended Reality (XR) arises as one of the current cutting-edge educational and entertainment emergent technologies. This service differs from traditional video streaming approaches due to its immersive experience, which allows the user to enjoy omnidirectional multimedia. However, the service experience must be guaranteed to avoid side effects such as cybersickness or disorientation. This work presents a framework to assess 360-video service streaming performance over mobile networks through Key Quality Indicators (KQIs) using VR (Virtual Reality) HMD (Head Mounted Device). The testbed is composed of a 360-video client for DASH (Dynamic Adaptive Streaming over HTTP), which playbacks multimedia content from a video server located in the cloud while KQI measuring tasks are performed in the user as on the network sides. Various metrics are collected such as resolution, frame rate, initial playback time, throughput, stall events, and round trip time (RTT), among various others. Finally, a performance comparison between LTE and 5G technologies is provided. Results from the KQI measurement highlight the potential of the new generation of mobile networks in the provision of service with high-quality levels of experience.**

***Index Terms*—Mobile communication, virtual reality, quality of experience, video, 360-video, key quality indicators, network performance, metaverse, extended reality.**

I. Introduction

THE MULTIMEDIA services are undergoing a complete revolution in their paradigms and characteristics. This development is due in part to the emergence of user devices with high computational capabilities, as well as to the emergence of new-generation access networks and transport infrastructures, and the fact that providers are bringing content closer to their end-users.

In this context, one of the technologies that has aroused great interest is virtual reality. The main objective of VR is to generate a sensory stimulus so that the user is able to experience a situation similar to physical reality, but in a simulated environment and in an immersive way. This added value has allowed VR to greatly surpass traditional multimedia services such as video streaming, remote meetings, and the development of key use cases like VR-gaming, revolutionizing how they can be enjoyed. One key application for immersive services with a great potential for VR is 360-video.

Immersive video, surround video [1] or so-called 360-video allows the user to play multimedia content in an immersive and interactive way. This approach consists of content recorded, by an omnidirectional camera or a group of those, in several directions at the same time, so producing a dynamic experience. In this context, the media can be viewed through a plain screen in a panoramic manner (e.g., YouTubeVR [2] for PC or smartphones), or via an HMD, where the user can modify its visual field using a natural head movement. In addition, this service can be deployed in an on-demand or live scheme. According to its architecture, the content may be hosted in a media server (e.g., CDN - Content Delivery Network) or be directly distributed from its source to the user premises [3].

This new paradigm lets professionals from several areas think about the different ways that this service can be exploited to generate new added-value services and experiences. Thus, multiple applications have been oriented to this service. One of them is virtual marketing applied to tourism areas [4]. Through this strategy, tourism agents can measure the degree of satisfaction or desire of users to visit a place. Other studies have made it possible to analyze the application of this service in education and training as an alternative to the traditional scheme [5], [6].

Despite the scheme adopted, deploying this service is challenging due to the need for production, transmission and presentation tasks, which are more complex than traditional 2D video streaming services. This implies that the multimedia content must be treated and delivered especially, so the user can feel within the content itself. However, acquiring 360-video is a procedure that may incur additional latencies due to the stitching and encoding tasks. Likewise, to generate a real-feel experience the frames must be formed by a higher number of pixels, and even higher frame rates,

Manuscript received 8 November 2021; revised 16 May 2022; accepted 10 July 2022. Date of publication 20 July 2022; date of current version 31 January 2023. This work has been partially funded by: Ministerio de Asuntos Económicos y Transformación Digital y la Unión Europea - NextGenerationEU within the framework "Recuperación, Transformación y Resiliencia y el Mecanismo de Recuperación y Resiliencia" under the project MAORI; Junta de Andalucía and ERDF (Consejería de Transformación Económica, Industria, Conocimiento y Universidades via Proyecto de Excelencia PENTA, P18-FR-4647) and Universidad de Málaga through the "I Plan Propio de Investigación, Transferencia y Divulgación Científica de la Universidad de Málaga". This work has been also supported by Junta de Andalucia through Secretaria General de Universidades, Investigación y Tecnología with predoctoral grant (Ref. PREDOC_01712) and postdoctoral grant (Ref., DOC_01154, PAIDI 2020); as well as by Ministerio de Ciencia y Tecnología through grant FPU19/04468. The associate editor coordinating the review of this article and approving it for publication was L. Skorin-Kapov. *(Corresponding author: Sergio Fortes.)*

The authors are with the Telecommunication Research Institute, E.T.S. Ingeniería de Telecomunicación, Universidad de Málaga, 29010 Málaga, Spain (e-mail: sppulla@ic.uma.es; jcbg@ic.uma.es; sfr@ic.uma.es; ebm@ic.uma.es; rbm@ic.uma.es).

Digital Object Identifier 10.1109/TNSM.2022.3192762

which increase storage issues. In the same way, transmitting this media over constrained networks may cause additional latencies (E2E communication, initial startup times), packet losses, freezing events, video quality fluctuations, and so on. In this fashion, the displaying tasks might suffer from decoding and rendering latencies, according to the user equipment, access technology, etc. or image distortion caused by the video projection on certain surfaces [3].

Although the 360-video service generates a huge interest among diverse users, a negative experience can generate the reverse effect. The way to quantify the grade of satisfaction can be defined from the objective and subjective analysis. In VR applications, the Quality of Experience (QoE) is determined through four factors: perceptual quality, acceptability, cybersickness and presence [7].

The perceptual quality represents the condition of the multimedia content, generally affected by the video resolution. Acceptability measures the degree of acceptance of the service, where the user assesses the desire to utilize the service. This last factor is most affected by resolution changes or freezing events, characterized by not clear content transmission [8]. Even though both factors are common for a traditional video service assessment, 360-video involves additional features due to its nature. On the one hand, the cybersickness factor evaluates the grade of dizziness produced while playing the content. On the other hand, presence is the analysis of the grade of the user's immersion within the content, which is intended to be produced using HMDs (Head Mounted Devices) [7]. Cybersickness is a subtype of sickness produced by the input of sensory stimulus that is not related to a physical movement which produces similar symptoms to motion sickness [9], [10].

Regardless of the quality factor, VR can lead to physical issues in the physical world. This is because a low-quality experience can cause adverse health effects such as motion sickness, confusion, anxiety, fatigue, or real-world injuries [11]. The cause of these problems can include insufficient image quality, optical distortions, lack of ergonomics in the device, or poor responsiveness due to delays [12]. In this scope, some of the HMDs vendors convey some of the must-do recommendations that are needed to assure proper safety levels while the products are used by users [13].

In this, the concept of motion-to-photon latency, which is the time between an action occurring in the real world and its respective representation rendered and presented on the screen, becomes extremely important. Previous works have shown that this latency should be less than 20 ms for an optimal experience and no more than 60 ms for acceptable service [14], [15]. Regardless of these time requirements, reaching low latency communications is a key challenge for live services. Several experiments obtain average E2E latency values of about 720 milliseconds in [16] for LTE networks using CMAF (Common Media Application Format) standard, and around 300 ms in UL and DL direction in [17]. Furthermore, on-demand services may reduce the E2E latency due to the media already hosted in a server, but they still have to face downlink delays.

This latency is a key issue for the development of online VR applications, or even for cloud approaches, that require the use of communication networks to download content, synchronize data or manage service-states between users, or provide added-value services. This fact adds delays due to geographical distance, as well as those generated by processing tasks and the network access technology, conditioning the experience in these types of scenarios. Despite this fact, it is expected that 5G and B5G networks provide specific conditions (e.g., 5QI) that support the deployment of this kind of service [18].

All these commented challenges are the key factors that encourage the authors to analyze the 360-video study case, its behavior under different communications network conditions and their implication on the level of QoE. To the authors' knowledge, the analysis of the quality of the VR experience, based on high-level metrics such as KQIs has not been previously performed. The KQIs are user-centric parameters that objectively represent the degree of user satisfaction with a service. These parameters allow quantifying the quality of service and relate the performance of the network that connects the user to their service provider.

In the same analysis the authors have done, no article qualitatively assesses the network impact of a 360-video service. This article provides a framework to evaluate the 360-video through a VR testbed. This allows measuring KQIs in the client equipment (HMD) as in the transport network. It is important to remark that this framework is network-agnostic, which allows the service to be assessed for various network technologies such as 6G and 5G, where XR is one of the top use cases, however, the tests were made over currently available technologies such as LTE and 5G.

The testbed is conformed by various iterative experiments, where the bandwidth and channel conditions are modified meanwhile the 360-video client runs on the HMD measuring and gathering network and service performance metrics. In the client premises resolution, frame rate, startup time, stall events, round trip time, buffer state, throughput, and resolution switches among various metrics are collected. The network reports throughput and several radio-related information. During the tests, the data is stored and managed using a RESTful server (Representational State Transfer). Then, this work also contributes with statistical analysis to compare and define the scenarios where 360-video service can be suitably deployed.

This paper is organized as follows. First, Section II presents some previous work related to a brief VR requirements description, media optimization and QoE estimation for VR services. Second, Section III describes typical architectures in virtual reality as well as the design process of the VR client. Next, Section IV details the configuration and deployment of a framework that enables KQI measurement for 360-video through a testbed. Section V presents the results obtained as a result of a set of tests using the developed system. Finally, we conclude with the most important points and challenges faced in this work.

II. Related Works

The requirements for VR applications and services must be adequate to establish an acceptable level of experience.

Research has been previously performed in several fields aiming to improve the way the service is delivered. Nonetheless, no former work has focused on the network and its impacts on the service as this paper presents. Research has been summarized here to point out some previous contributions in this field.

This section has been divided into two subsections. Firstly, the *VR requirements* Subsection presents the state-of-the-art conditions aiming to reach suitable 360-video service, which supports the results of this paper in Section V. Secondly, Section *Quality of Experience* presents a summary of varied works focused on analyzing the QoE.

A. VR Requirements

XR is a technology that seeks integrated experiences mixing real and virtual elements and environments. This scope involves VR, AR (Augmented Reality) and MR (Mixed Reality). The degree of interaction with each reality is what defines the specific technology needed. Moreover, the idea of offloading computation loads at the user devices and providing more mobility (currently constrained by high-quality HMDs that use wired connections) strengthens the wireless networks as a possible solution to deploy these cutting-edge services.

On the one side, MR (Mixed Reality) and AR (Augmented Reality) are specific categories that overlay virtual elements, and with the real environment. MR merges some physical reality elements such as the player silhouette in a virtual reality environment. Conversely, AR overlaps some virtual elements and information over physical reality. Meanwhile, VR provides a whole virtual experience.

For AR and MR cases, computational capacity is required to calculate three-dimensional models that allow finding the adequate position of the overlay elements in order not to disturb the user's field of view and its sensations. In some cases, uploading video may be necessary to offload local computational resources and do model calculations, element recognition, and so on a remote server. However, these actions may stress uplink communications.

Conversely, VR needs much more computational resources to generate whole virtual environments and generate a truly immersive experience. Commonly, VR services require downloading or streaming multimedia content to a VR-ready device, which is an appliance prepared for VR rendering (two screens being rendered at the same time). According to [19], a *real-feel* resolution could be reached with about 60 pixels per degree with a 120 Hz frame rate, based on human eye perception close to 120 degrees FOV (Field of View). To reach these conditions an approximate bitrate of 1 Gbps is required for a theoretical per-eye 9K resolution. However, this is not reachable by current technology. For instance, today's 360° 4K camera streaming at 30 FPS may generate up to 300 Mbps of data [20]. Dasari *et al.* [21] mention that the 360-video service needs to download about 8x more content than the traditional version (4k per eye in comparison to the typical 1080p ratio), which supports the fact that this service requires adequate network resources.

The authors in [19] also point out that reduced latency is mandatory to assure media smoothness avoiding virtual and motion sickness. As previously stated a 20 ms latency [14] is recommended for this kind of service. Nonetheless, current mobile networks can not face these requirements (LTE typical loopback latency is about 25 ms).

Some approaches have been proposed in previous works to improve VR performance in terms of media content (frame tiling) as network implications (distributed computing). These both are intended to reduce the overall latency and the throughput needed. The work in [19] presents a VR gaming application that uses MEC computation and caching. Moreover, Siriwardhana *et al.* [22] describe several architectures that may be exploited by 5G mobile technology. In the review, local, edge, cloud and hybrid architectures are considered. Bringing computing processes close to the user can significantly reduce latencies.

B. Quality of Experience

Given the importance of quality of experience, its measurement has generated high interest from researchers. The authors in [23] present an analysis of metrics that allow quantifying image quality in a 360 video, with PSNR (Maximum Signal to Noise Ratio), WS-PSNR (Weighted Sphere), among various. The authors state that PSNR is the most suitable metric for 360-video due to its low complexity. Hanhart *et al.* [24], analyze objective video quality using WS-PSNR and subjective metrics using MOS (Mean Opinion Score). The authors assess four different sequences and how codecs can affect the visual quality after the encoding and decoding processes. Nevertheless, these metrics assess the quality in terms of the image quality itself but are not related to the service perceptions from the users as our work does.

In addition, Filho *et al.* in [25] presented a platform called VR-EXP which is intended for the evaluation of 360° video streaming performance. This platform is composed of a central VR video client emulator, which measures the playout performance based on various ABR schemes, tile-based mechanisms and data-driven network conditions, and a Network Performance Enforcement Point that exploits the balance of flexibility and accuracy for experimentation. This module is SDN-based (Software-Defined Networking) and emulates network conditions based on input datasets. Taken together, this approach offers a good quality evaluation of the service, however, the framework to be proposed in the present work aims to represent the network impact over the service in a real radio scenario for mobile networks, where the influence of other radio technologies degrade the quality of the channel. In addition, this work assesses the quality of the service through KQIs measured in a real HMD, where the nature of VR technology can be fully experienced as well as affected by the network and user equipment's real limitations, which is the key point of this contribution.

The work [26] developed by Anwar *et al.* presents the estimation of Quality of Experience using a Bayesian model with stall events and video bit rate as inputs. To develop the model, a subjective assessment was made in order to obtain statistical data from users' perspectives. A different approach is done by Park *et al.* [27], where the authors presented a mechanism to stream adaptive video tiles based on Neural Networks (NN) to predict the user viewport avoiding transmitting the entire 360-video. In the same work, another NN strategy is approached to adapt the video rate. The QoE is estimated using objective metrics based on the video bitrate and the overlapping tile ratio, its variation, miss ratio and rebuffering. The tests were made over WiFi and LTE, however, no network-related analysis was provided as well as in other previous works.

QoE modeling from network information and KPIs (Key Performance Indicators) is another important topic that is being assessed by researchers. The difficulty in objectively estimating the QoE in an E2E (End-to-End) service is due to the operators' inability to access high-layer data. The security of the user devices as well as the privacy of their data and the appearance of new services make it difficult to gather QoE data.

In this context, the work of Herrera-Garcia *et al.* [28] proposes an ML-based estimation of KQIs through low-layer data (e.g., KPIs) for FTP service. In the same research line, Baena *et al.* in [29], describe a methodology for ML-based KQI estimation for video streaming. The authors propose a framework for network slicing negotiation for verticals in a 5G context. These contributions define a baseline for automatic network management based on the quality of service provided.

With regards to KQI acquisition, Peñaherrera-Pulla *et al.* [30], present a framework to gather high-layer metrics for Cloud Gaming service through wireless networks (WiFi, and LTE) using a testbed that iteratively executes experiments. The input parameters are video resolution, frame rate, and transport network technology among others.

A key work in this area is presented by Krogfoss *et al.* [31]. The authors propose a novel strategy to estimate QoE through a multiplicative model based on the impact of case-specific KQIs for 360-Video and VR-Gaming. Likewise, the authors describe a methodology to map KQIs to KPIs in order to approximate quality metrics based on RAN records. The model is tested on LTE and 5G cloud approaches. Conversely, our proposed work aims to analyze and qualify the QoE based on real-time KQIs estimation among several channel conditions, which can provide a precise perspective on how the network can play an important role in the future deployment of this kind of services in 5G and B5G networks.

III. CLIENT DESIGN

The *client* allows connecting the HMD with the server but also requesting, decoding, rendering and presenting the video frames in an omnidirectional manner based on the area of interest determined by the user's viewing angle. Likewise, this development allows obtaining specific streaming video metrics, which quantitatively reflect the quality of experience of the service. This implementation is hardware-agnostic, so it does not depend on the used HMD.

The details of this development are pointed out in this section as follows. Firstly, a theoretical VR-architectures approach is provided, then the application designing process is stated. Finally, the metrics measurement tool is described.

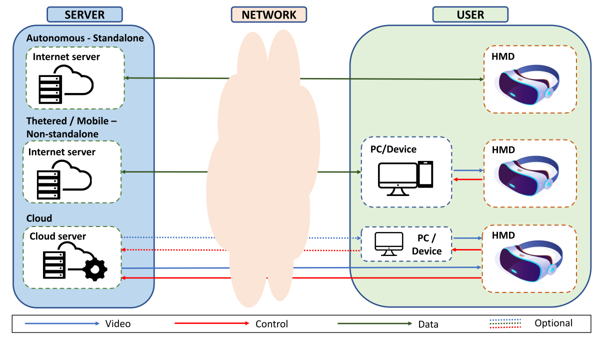


Fig. 1. Typical VR HMDs architectures.

A. Virtual Reality Architectures

The are several classifications for VR technologies based on different concepts such as user behavior (visual, haptic and multisensory devices), their way of detecting space or movements (tracking), or the physical connectivity of the virtual reality HMDs. The authors decide to focus on the latter classification because is the closest to the work objectives and it is the best defining on how commercial devices are currently being offered to the market.

The physical connectivity category mainly highlights the way in which VR HMDs or simply HMDs are able to operate in dependence on other equipment for the use of hardware or software resources. The most commonly used schemes are Standalone, non-standalone or tethered, and cloud-oriented schemes. This classification has been summarized in Figure 1.

Standalone HMDs are those that have the necessary hardware and software resources to generate an immersive experience. This means that the devices are capable of processing, rendering and presenting omnidirectional multimedia content without the need for other supporting equipment; however, their performance may be limited. Complementarily, they can connect to external networks to run specific tasks and services.

The *tethered* or *non-standalone* devices have limited resources to provide a virtual reality service. In most cases, they require the help of a support device, usually a computer or a cell phone, which takes care of the processing and rendering tasks. In this way, the content delivered to the glasses is a multimedia stream encoded and optimized for playback on the VR equipment [32], [33].

The development of next-generation network infrastructures, in conjunction with edge and cloud computing paradigms, has enabled the concept of *Cloud VR*. The key idea of this architecture is to delegate computationally demanding tasks to equipment with large processing capabilities, thus making it possible for the user equipment to be hardware-light. In this way, the content is processed and rendered on the edge devices and then sent over the network [34].

The main disadvantage of this concept is the additional latency due to the geographical distance between the servers and the user. Solutions that can be addressed to overcome this

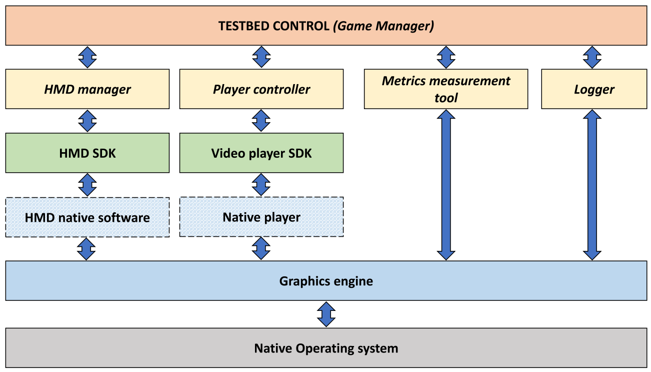


Fig. 2. Implementation stack model.

include: the implementation of services with edge computing, the use of low latency access networks as those supported by 5G technology. In addition, low latency codecs can also be considered, so that the total latency does not exceed the mentioned limit (20 ms for optimal experience to 60 ms in the worst case) [34].

The developed work presented in this paper, implements the 360-video service using a standalone scheme for its VR HMD. The client design and its implementation are approached in the following subsections. In addition, Section V describes the testbed setup that uses this client to generate some controlled tests.

B. Application Development

The video client was developed as a high-layer implementation through Unity Engine as the graphic engine that holds up the application. The supported functions include managing and controlling tasks as well as the graphical processes such as video decoding, rendering and presentation process, as well as audio decoding and synchronization.

The integration of the application with the VR HMD environment is done by means of the HMD SDK (Software Development Kit). This set of tools provide an API (Application Programming Interface) that enable the use of external scripts and serving as bridge with the native language of the HMD. This mechanism lets the control of the system in an autonomous way, as well as access to certain local resources such as sensors, hardware information and event usage.

At the software level, the application is made up of four modules that manage its integral operation. Each of them fulfills specific control and management functions. This scheme follows the layered model presented in Figure 2.

The application is controlled by a *game manager*. This object is responsible for managing the execution of the object that performs the measurements, as well as the object that controls the media (video + audio) and the one that collects the data obtained and records them in an external file. The game manager also oversees the events generated by the HMD tracking system. This allows following the movement of the user's head, and thus presents the multimedia content according to the user's field of view. Although the media is presented in a specific FOV, Unity renders the whole scene (360°) per screen update. This is due to that the downloaded content is entirely transmitted (360° sphere) from the server using a non-tiled strategy. In addition, it is remarkable to point out that the HMD processes the content and renders it in a two-eyed manner, so the user can visualize this content in 3D through the HMD stereoscopic display, allowing the user to perceive depth stimulus. These tasks are controlled by the *HMD manager* and informed to the *game manager*.

Another module is the *player controller*. This object is responsible for managing the functions of requesting the resource via the Internet, playback, and closing the resource, among others. These methods have been integrated, as an upper layer over AVProVideo player SDK in its free trial version available on GitHub [35]. This allows the multimedia controller to be centrally managed from the *game manager* in the Unity layer, while the player built over this SDK is only responsible for visualization functions on HMD's screen (decoding and rendering). Moreover, this SDK includes ExoPlayer libraries [36], an open-source extensible Android-oriented video player which offers features that are not available in the native Android player such as a DASH format (opening and closing media, requesting segments, etc.).

Finally, the *logger* and *metrics measurement* modules have specific functions in the application. The *logger* is responsible for updating and storing the measurements obtained during the execution of the application. This block writes the measurements obtained during a specific number of display updates into a Unity object. This object allows the client to transform its information into a json file, which will be transmitted to the REST server when the *game manager* finalizes an experiment iteration. By default, the value of 72 FPS (Frames per second) has been set, which coincides with the default value of the display update rate of the VR glasses. In this way, the measurement of parameters for each second of execution of the application is obtained in a precise way.

C. Metrics Measurement Tool

This module is composed of a set of tools programmed in C# and integrated into Unity Engine for the measurement of indicators that represent, in a quantitative way, the quality of the service.

According to the literature, the QoE assessment through KQIs is service-specific, which means that each service gives priority to certain factors (video quality, audio, latency, ...). The 3GPP stated in [37] that mobile video streaming service quality may be reflected via accessibility (start delay, start success ratio) and integrity (stall frequency, stall time ratio, downlink throughput) metrics. Likewise, Krogfoss *et al.* [31] point out that the typical KQIs for video streaming are related to frame rate, resolution, stalling and latency. However, latency is not a crucial factor in cases where the interaction and the responsiveness are limited as in 360-video use cases. In this same context, additional metrics such as the throughput in the client, the number of resolution switches, the state of the buffer and the round trip time can provide a better understanding of the network impact on the service.

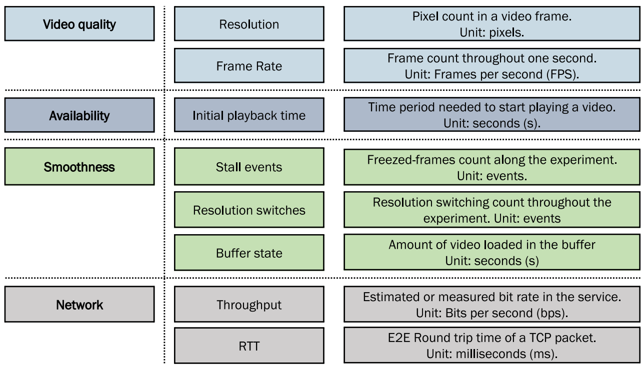


Fig. 3. 360 Video Key Quality Indicators.

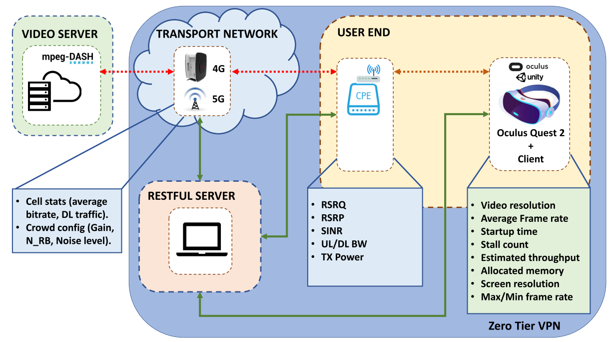


Fig. 4. Implementation scheme.

The developed client tool allows the measurement of the mentioned indicators: resolution, frame rate, number of freezing events (stalls), throughput, RTT, buffer state, resolution switches and several statistics related to the graphic process in VR HMD such as maximum and minimum screen refresh rates, memory usage, equipment data such as operating system, maximum resolution, average video stall time, among others. The key indicators that have been taken into account in this analysis are pointed out in Figure 3.

This tool does some measurements per second, depending on the HMD's screen frame rate (hardware). Nonetheless, the video resolution and video frame rate are metered just once per second. The stall events are counted for each HMD stalled frame, then the average inter-frame time is calculated for each experiment. This allows the tool to estimate the total stall time by multiplying the stall frames by the frame period.

The initial playback time is calculated once per iteration. This time starts when the video client requests the DASH manifest. Then, this parameter is timed until the client receives enough video frames to fill the buffer up and the playback starts. Additionally, the buffer health is also calculated, offering an overview of its state throughout the entire session. To do this, the framework takes the difference in timestamps of the available media segments which have been downloaded into the buffer.

Complementary, the network performance is also assessed in the HMD via the throughput and latency. The throughput is measured using the segment size and the time employed by the content to be downloaded into the buffer. In turn, network latency is estimated through the RTT. This metric considers the time between a packet is sent to the server and the response that arrives back to the client. This measurement is done every second through an additional software thread in the HMD client using TCP (Transport Control Protocol), which is the transport protocol used for DASH.

IV. Testbed Implementation

The implemented framework comprises the elements depicted in Figure 4. On the one side, the video client (see Section III) lets the multimedia content be displayed in the HMD as well as gathers some performance and service metrics related to 360-video QoE. All the measurements are temporally stored in the device for each experiment.

On the other side, the multimedia content is downloaded from a server located in the cloud, which stores a 360-video source in conventional DASH-MPEG format [38]. This work considered a 360° non-tiled multimedia source in DASH format instead of tile-based content in order to analyze the actual effects of the network on the service without any kind of spacial optimization. The use of optimization mechanisms in the source reduces the network stress which is not the key case study of this work.

The player for its part supports HLS (HTTP Live Streaming) and DASH formats for ABR (Adaptive Bit Rate). These streaming strategies can be used in order to reduce the video quality (resolution) when the channel conditions are not proper or modify compression levels. This allows assuring a stable playback, reducing stallings, and keeping a steady frame rate. The adopted ABR mechanism for the experiments is quality adaption through a player native throughput-based strategy. The available resolutions and their respective coding bitrates are described in Table I. In addition, the video buffer is configured to store 50000 ms of content and 5000 ms as the minimum initial buffer before starting playback. Furthermore, no viewport-aware rendering is used, since the whole sphere is streamed from the server, as previously mentioned.

Regarding the transport network, it aims to communicate the user equipment with the content server. This segment is featured by mobile networks (LTE and 5G NR) that will be defined in Section V. The HMD used in the system gets Internet access via a CPE (Customer Premises Equipment) that provides WiFi 6 connectivity and is linked to the transport network.

To centralize the data gathering process, the system integrates a *RESTful server* deployed over a flask server, which allows the HMD metrics to be hosted after every experiment. This server also requests some other network-related metrics (RSRQ - *Reference Signal Received Quality*, RSRP - *Reference Signal Received Power*, RSSI - *Reference Signal Strength Indicator*, SINR - *Signal to Interference plus Noise Ratio*, BW - *Channel Bandwidth*, …) from the CPE through HTTP-based commands through an API. RSRP measures the linear average power received for all the REs (Resource Elements) contributions that carry Reference Signals. RSSI denotes the overall received power in a specific bandwidth. For

TABLE I
TESTBED CONFIGURATION

Parameter	Description	Value
Iterations	Number of samples/measurements per experiment	120
Experiments	Number of experiments with the same configuration	60
Technologies	Transport network technologies to connect the user equipment with the Internet	4G/LTE 5G
Crowd-BW	LTE Crowdcell channel Bandwidth	5 MHz 10 MHz 15 MHz 20 MHz
5G-BW	5G channel Bandwidth	50 MHz
MaxPT	Crowdcell maximum power transmission level for *MaxPT* tests	0 dB
MinPT	Crowdcell minimum power transmission level for *MinPT* tests	-10 dB
RedPT	Crowdcell power transmission level for *RedPT + Noise* tests	-20 dB
Max-Noise	Maximum noise level for *RedPT + Noise* tests	-20 dB
Min-Noise	Minimum noise level for *MaxPT and MinPT* tests	-30 dB
Video resolution	Available resolutions for the video content at the server	720 × 360 1080 × 540 1440 × 720 2160 × 1080 3840 × 1920
Average bitrate per segment	Average bitrate per each video segment (same order as resolutions)	1 Mbps 1.5 Mbps 3 Mbps 5 Mbps 9 Mbps
Frame rate	Frame rate at which video is encoded	30 FPS
Segment duration	Time period for each video segment	4 seconds
CODEC	Video CODEC used	avc1.42c00d
Initial buffer	Filling time for initial playback	5000 ms
Min. & Max. buffer threshold	Minimum and maximum thresholds for buffer	50000 ms
Streaming protocol	Protocol used for streaming of media	Standard DASH
ABR strategy	Adaptive Bitrate strategy used for buffer filling	Throughput-based
HMD Model	HMD model used in the testbed	Oculus Quest 2
Operating System	HMD operating system	Android OS 10 / API-29
Graphic engine	Graphic engine that supports the video 360 client	Unity 2020.2
Video player	Video player API integrated in the video client	ExoPlayer NonOES

its part, RSRQ is a derived measure from RSRP and RSSI that indicates the received power quality along all the REs. SINR refers the ratio between the desired signal and the undesired noise and interference.

It's remarkable to mention that a virtual network was configured only for data management using *ZeroTier* [39]. This tool allows the system to be virtually interconnected, even though each element would use a different technology (WiFi, mobile network) or IP network. In addition, after measurements of the round trip time (using ICMP protocol), it was possible to establish that the average latency between the user and the media server [38] was about 35 ms for the 5G network and 48 ms for the LTE technology. Nonetheless. this is a relative value because there is no current streaming, therefore, the network is not stressed.

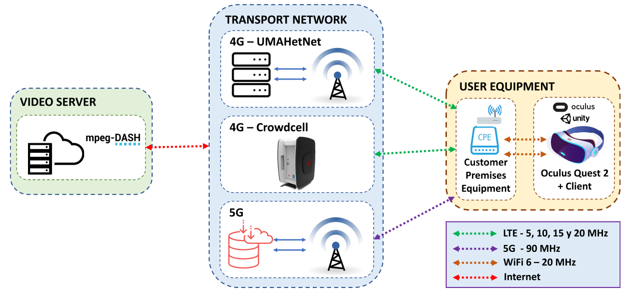


Fig. 5. Testbed Setup.

The communication loop starts with the HMD requesting the multimedia content to the video server. Then metrics are calculated and stored while the experiment finishes. Once the iteration is done, the restful server receives the metrics gathered in the HMD (video resolution, frame rate, ...), and then, the request statistics to the CPE. The obtained data is locally stored in the server in JSON (JavaScript Object Notation) format.

In the LTE case, extra information is provided due to the use of an LTE Crowdcell as the transport network [40], [41]. This element is an open-source solution that implements SDR (Software-defined radio) and a virtualized LTE-core designed for strengthening the coverage in indoor environments. The Crowdcell gives the system the capacity to be flexibly modified in terms of channel bandwidth, noise presence (SDR-emulated), transmission power, etc. Besides, this network element is also connected to the management network through ZeroTier.

V. EVALUATION

In this section, we present key results obtained with a testbed that deploys the 360-video service using the system described in Section IV. This framework tested the network impact on the service through KQIs measured on the client and network sides. The transport networks used are an LTE Crowdcell (Network-in-a-box), an LTE commercial-like network deployed on the University of Málaga campus and a 5G pilot network. The CPE provides WiFi connectivity for the HMD, allowing this to reach the Internet through the mobile network. The schematic architecture of the scenario is depicted in Figure 5.

A. Crowdcell Tests

The results obtained in this subsection belong to the testbed configured with an LTE-Crowdcell device acting as the transport network.

On the one side, the experiments were configured with different bandwidth channel combinations (5, 10, 15 and 20 MHz). Also, parameter-tuning was made in order to control the transmission power (gain parameter). Noise was emulated using an SDR Crowdcell's tool (noise level parameter). These combinations allowed to deploy scenarios with Maximum transmission power (*MaxPT*), Minimum transmission power

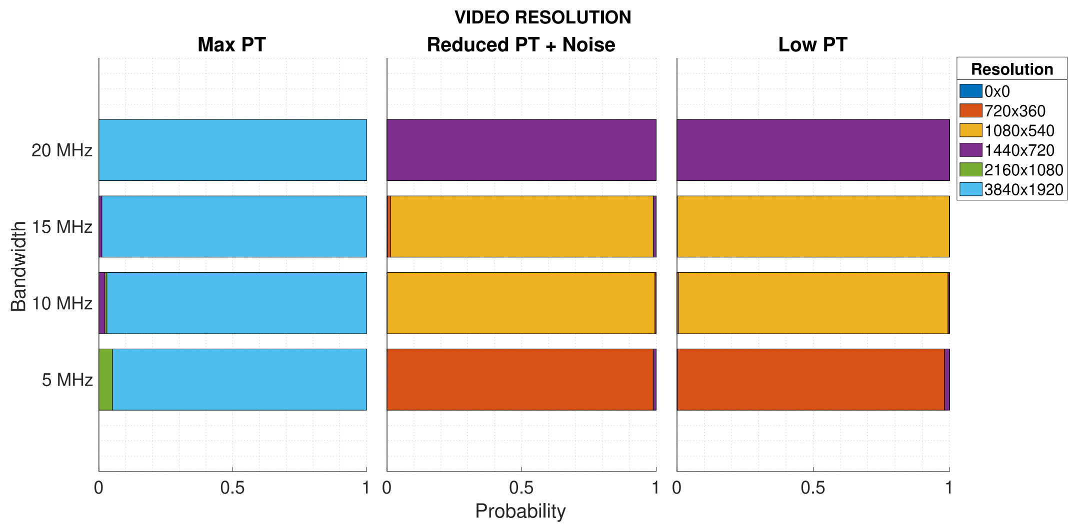


Fig. 6. Video resolution distribution per Bandwidth.

(*MinPT*) and a special case with power reduction and additive noise (*RedPT* + *Noise*). Each one of the aforementioned cases is tested with all the available channel bandwidths.

The maximum value of transmission power belongs to the maximum value configurable for the Crowdcell. Conversely, the minimum value of power corresponds to the minimum value possible that prevents losing the connection between the Crowdcell and the CPE. In this same context, the noise levels were tuned in order to reduce to about a quarter of the SINR at the first channel condition. This aims to experimentally represent a channel in normal conditions, a channel with reduced power and interference, and a channel affected by interference and considerable attenuation. The summary of the most important parameters in the testbed can be checked in Table I.

On the other side, the video client was configured to perform multiple and iterative experiments with a duration of 120 seconds. In the same context, each experiment gathers information related to the service quality (resolution, frame rate, ...) through measurements in the HMD as in the transport network and the CPE. Besides, it is remarkable that in all the tests the media is requested from the server and deleted from the memory at the final of the iteration, this way, there is no buffered residual information. With this consideration, it is possible to ensure that each of the tests is only network-dependent and the metrics can be objectively analyzed.

The results in Figure 6 show the behavior of the *video resolution* when tested with different LTE channel bandwidths and channel conditions. It is remarkable that all the channel combinations, even with a limited 5 MHz channel, with high transmission power have presented the media content for the most time with the maximum possible resolution available. Nonetheless, this is not practical for real cases where cell-edge users may face adverse channel conditions.

In the second case in Figure 6 the transmission power was reduced and noise presence was emulated in the Crowdcell. The results indicate that the service may suffer from QoE depletion due to the reduction of the resolution along the cases. The most affected case is developed by the 5 MHz, which reaches the minimum available resolution in the source. Despite the worst case, it is important to notice that even the 20 MHz case bears a reduction in quality compared with the previous scenario.

In the third case, the transmission power was set to the minimum possible in which the radio link is available. The outcomes exhibit that the most restrictive channel is not suitable for a high-resolution service. In fact, the performance depicted is very similar to the former channel scenario.

To complement the information provided by Figure 6, the number of resolution switching events is depicted in Figure 7. This shows that in all the cases, the trend is to keep the resolution such was initially loaded. This means, for instance, that the high-quality channel experiments maintained the 3840x1920 resolution throughout the 120-second experiments, while the low-quality channel ones used 720x360 along theirs. Although, this Figure also points out that certain experiments changed their resolution even up to 3 times (5 and 7 switching events are outliers that only belong to one experiment).

Coupled with the resolution, the *frame rate* provides more information about the video quality in the 360-video service. As displayed in Figure 8, the maximum power case reveals that all the channel bandwidths are able to keep a stable service with an average video frame rate of 30 FPS despite the displayed resolution.

In contrast, the 5 and 10 MHz channels show affected displaying rates with respect to the 15 and 20 MHz. In this context, the player is intended to adjust its current frame rate in function of the number of available decoded frames. This factor depends on the state of the buffer. If some number of frames is accessible, the playback may stall, in the worst case, or modify its displaying rate in order to compensate for the missed frames.

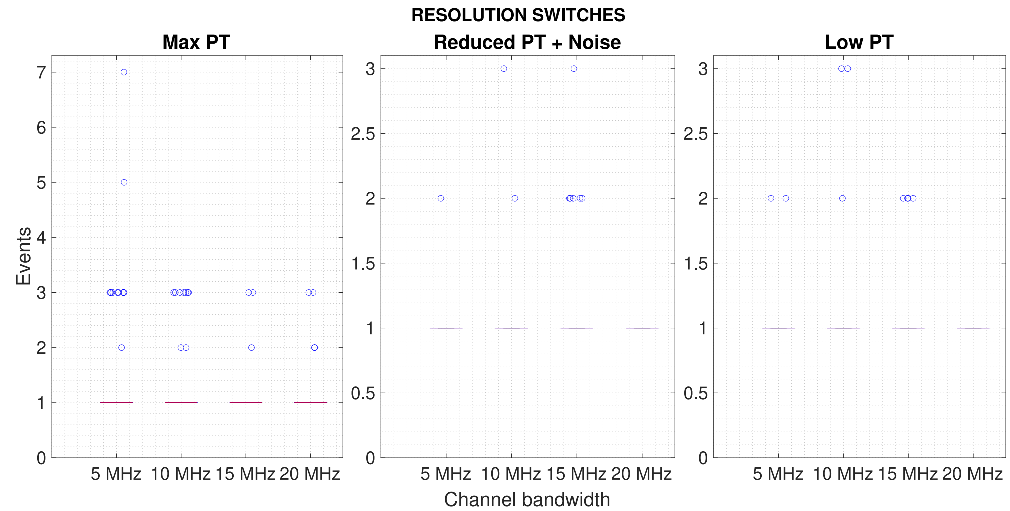


Fig. 7. Video resolution switching events per Bandwidth.

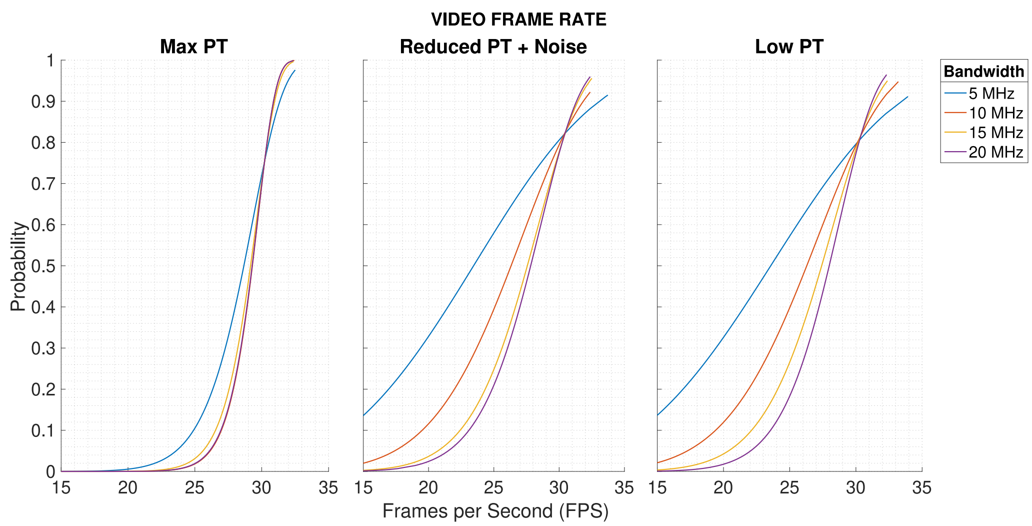


Fig. 8. Video frame rate CDF per Bandwidth.

In this context, the improper channel conditions (attenuation and noise or interference) in conjunction with a bandwidth-constrained channel may importantly affect the quality of this service. As a consequence, the service will present a lack of smoothness producing effects like frame freezing that in VR service may produce dizziness and disorientation.

Furthermore, this analysis can be enriched through the information gathered for the stall events along the tests. The results in Figure 9 are a consequence of the network stress based on the channel bandwidth and its conditions. The stalling time calculation is done for each experiment by summing the number of stalled frames in the player and dividing by the total number of frames. Then, the time percentage values are calculated through the whole testbed experiments.

The results specify how the stall percentage remains close to zero for all the channel bandwidths while the channel conditions are suitable. This fact matches with the stable frame rate in the same experiments. Conversely, when channel conditions worsen, the stall time percentage increases for limited channel bandwidths like 5 and 10 MHz (in both channel scenarios). Again as expected, this indicator matches with frame rate and video resolution fluctuations in Figures 6 and 8.

Although the results leverage the idea that there are stalling events during adverse conditions scenarios, this does not imply that every experiment may suffer from it. To clarify this thinking, the stalling distribution is presented in Figure 10. The results show that the average stall event lasts about 2.4 and 2.8 seconds for the 5 MHz channel respectively each scenario. Regarding the 10 MHz channel, the average stall is about 2.4 s and 4 s. It is curious the presence of stalling events for 20 MHz channel, however, the logical thinking for this situation is that this represents only an isolated experiment (similar to 10 MHz in the same scenario) because of the steep slope in its CDF (Cumulative Distribution Function).

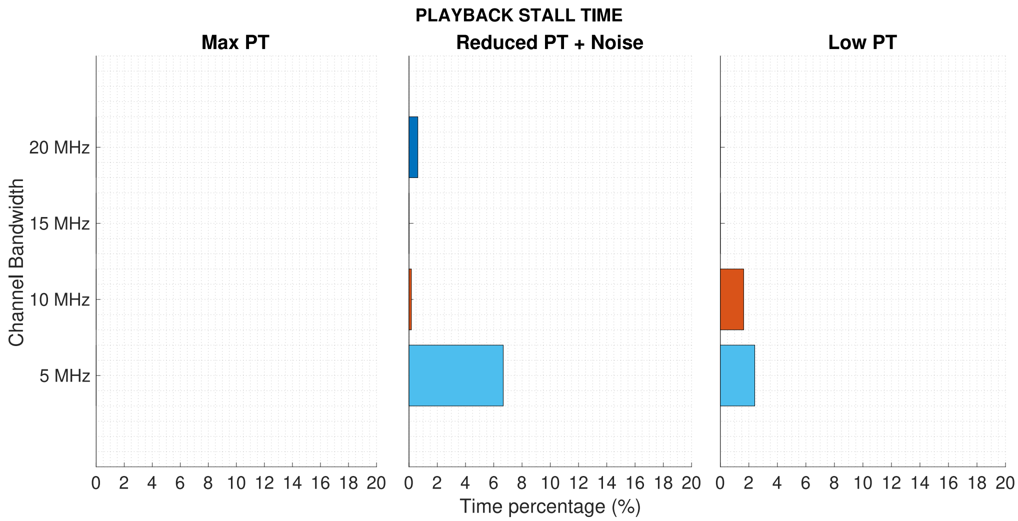


Fig. 9. Playback stall time percentage per Bandwidth.

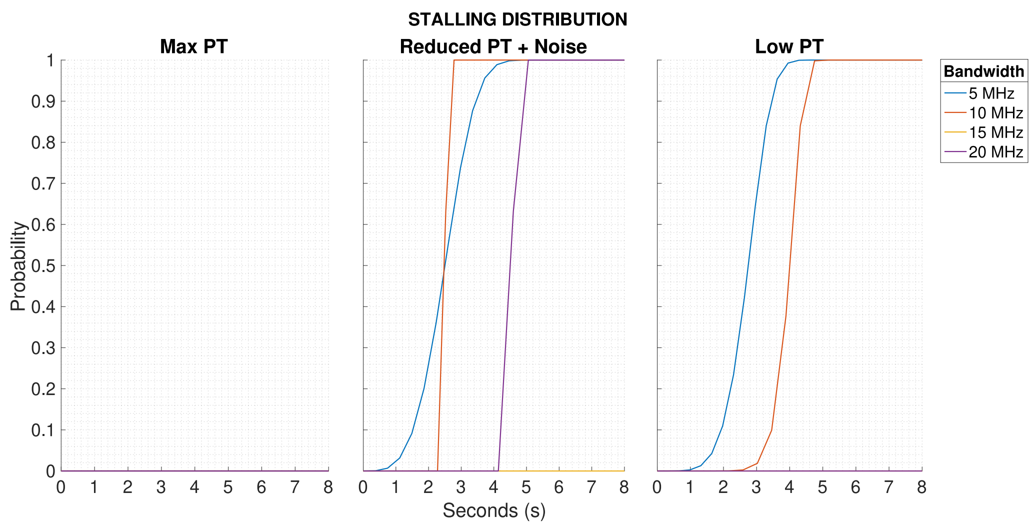


Fig. 10. Playback stalling distribution per Bandwidth.

A different perspective of the performance behavior of 360-video can be obtained with the *buffer state* information. This metric reflects the level of fullness of the buffer while the video is being displayed on the HMD screen. In order to summarize the details, the 25th, 50th (bold line) and 75th percentiles have been drawn. As can be seen in Figure 11, the channel scenario with high power transmission displays a good buffer performance, where it tries to keep a 50-second video reserve throughout the playback. Also, it is possible to observe that when the video is about to be fully downloaded, the buffer level only reduces (which is expected as a normal buffer profile).

The key interest of this metric is the information contained for the worsened cases. As shown, the buffer performance for 5 MHz displays a poor downloading pattern for both cases not being able to reach even more than 30 seconds of video. This impacts the increase of stalling events throughout the playback as presented in Figures 9 and 10. Conversely, the buffer profiles for the other bandwidths are expected. For instance, the 20 MHz develops worse than 15 MHz because this experiment downloads bigger video frames (3840x1920) with respect to 15 MHz (1080x540) in an affected channel scenario. Despite the buffer filling rate reduction, the 20 MHz still performs very well, presenting a high-quality video with a steady frame rate.

Likewise, the *Initial playback time* refers to the period of time needed to start the video displaying in the HMD. This time depends on the quantity of data required to fill the initial buffer. Indeed, this also depends on the video resolution and the channel capacity to transport its data. As presented in Figure 12, the initial playback time linearly decreases in terms of channel bandwidth. As expected, the 5 MHz channel exhibits the most restricted performance for both adverse channel conditions. It is remarkable that the maximum power case shows similar behavior for all the combinations.

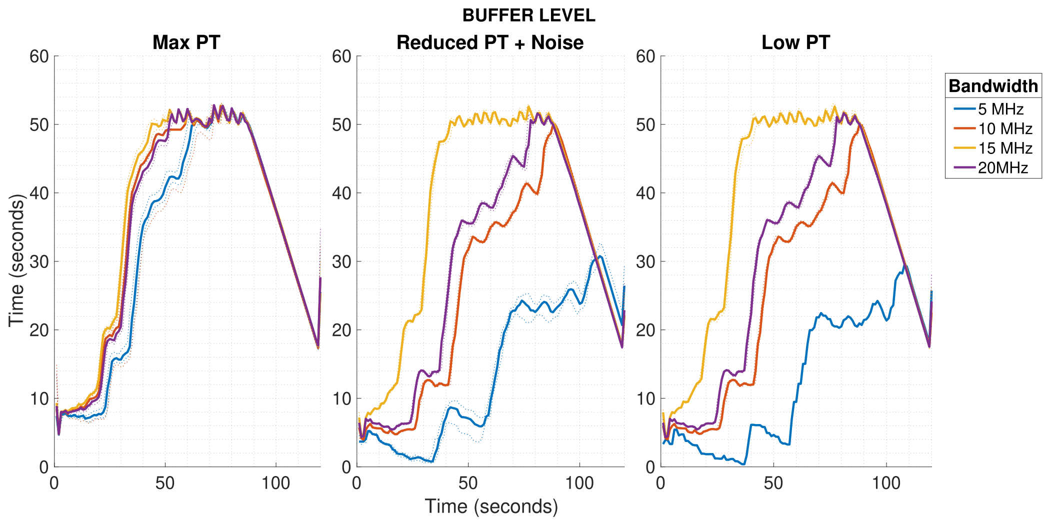


Fig. 11. Buffer state per Bandwidth.

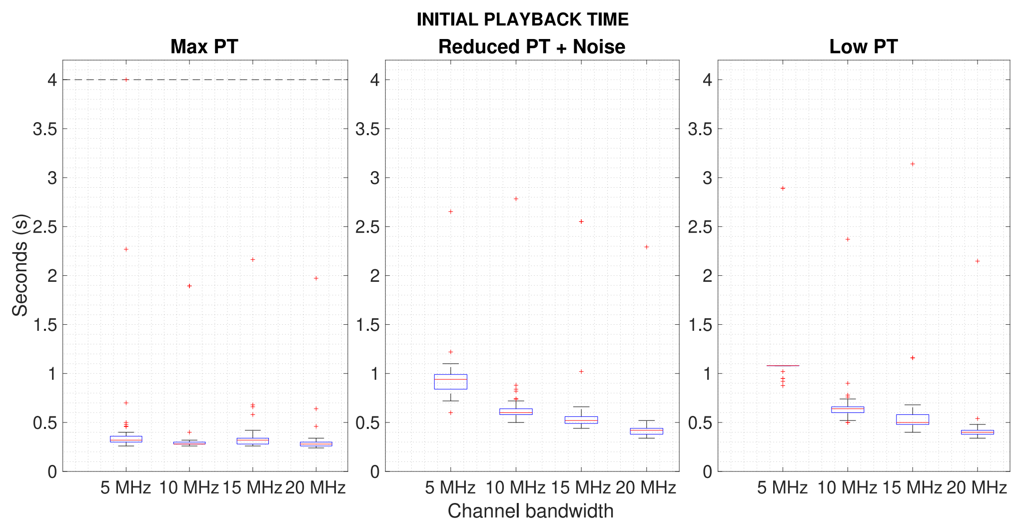


Fig. 12. Initial playback time per Bandwidth.

The above indicators are complemented with the throughput estimations done in the transport network through the Crowdcell. This measurement is represented as the average throughput per session in order to get rid of high variance values that are normally present in this metric (see Figure 13). As corresponds to the maximum power case, all the channels exhibit a similar behavior that matches with gathered results for the video resolution, frame rate and stall time indicators. A great difference in throughput is noticed when the channel is limited in bandwidth and radio conditions, as depicted in noise and low power cases for 5 and 10 MHz channels. As expected, the throughput profile follows an incremental pattern in function of the channel bandwidth used.

This pattern can be correlated with the throughput profile measured in the HMD by means of the client. As was anticipated, the average throughput per session in the maximum power scenario displays a similar performance for every bandwidth. In this same context, the incremental fashion of the throughput is repeated for both worsened cases. The difference in the average throughput is due to the change in the video resolution used. The bigger resolution is downloaded, the bigger the frame sizes are. This analysis comes from Figure 14. It is important to mention the difference in the measured throughput for the Crowdcell and the client owing to traffic generated by retransmissions.

The throughput measured in the Crowdcell provides a wide perspective of the service performance in terms of the bandwidth used for each channel. Nonetheless, the service behavior can be also characterized by the round trip time profile for each experiment. This metric has been measured using TCP traffic between the client and the media server. As shown in Figure 15, the RTT presents a similar average value of latency of about 60 ms for all the bandwidths used in the first scenario, nonetheless, this conduct is different for the worsened scenarios. As depicted, the latency tends to decrease as the bandwidth increases, ranging from values between 120 ms and

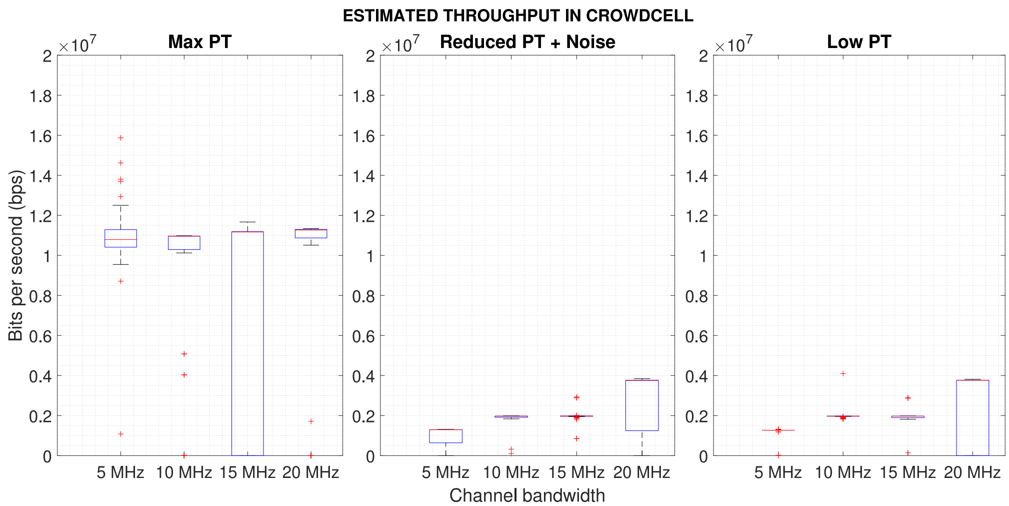


Fig. 13. Estimated throughput in Crowdcell.

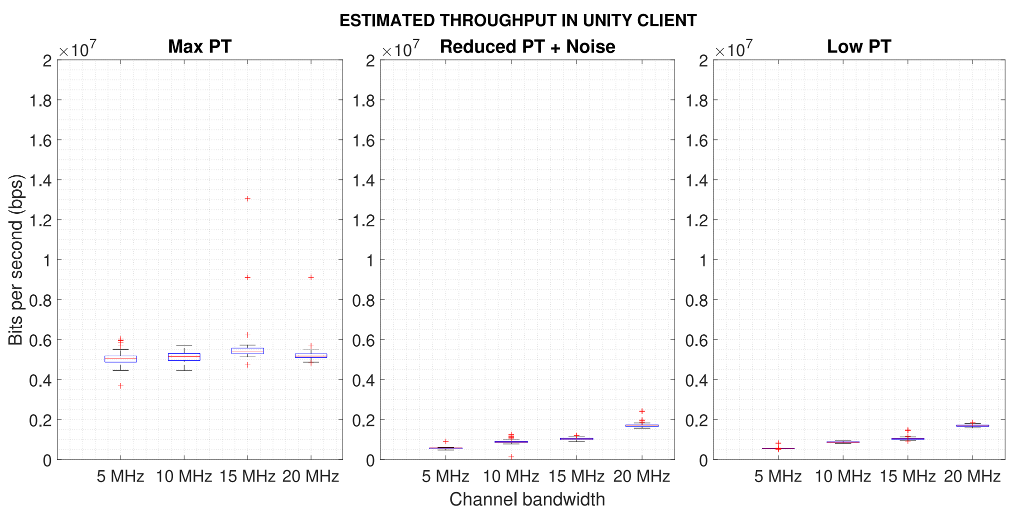


Fig. 14. Estimated throughput in Client.

100 ms on average. Although these results comply with the logical expectations, it is crucial to note that the latency in 20 MHz is a product of higher quality resolutions than in 5 to 15 MHz cases.

To finish the analysis of Crowdcell results, the number of retransmissions in the downlink direction has been included in Figure 16. This metric shows that a seamless connection between the user and the server existed for the maximum power case. Nonetheless, when the conditions worsen, the number of retransmissions increases as the channel bandwidth decreases. This provides a key insight into this service taking into account that the wider channel cases uses better resolution quality video.

Finally, in order to correlate this analysis of the results obtained along the experiments, Figure 17 depicts the RSRP measured for each scenario tested in the proposed work. As expected, the power received trend is aligned with the values configured for gain and bandwidth. The higher the bandwidth channel, the higher the power distribution along the spectrum used. If the power perceived by the CPE is worsen, the chances to be affected by noise are higher, which traduces in not proper channel conditions, therefore a degraded QoE of this service.

B. 5G Comparison

In this section, a 5G approach has been adopted in order to compare it with the previous analysis. The transport network has been modified to use the University of Malaga (UMA) LTE Picocell network (UMAHetNet) [42], [43], which is deployed and working in a real scenario. Moreover, this testbed was also tested with a 5G pilot network for research. This network disposes of outdoor and indoor cells installed at the University of Malaga campus. Likewise, for objective testing purposes, it is only used the indoor cells in order to get comparable results with the picocell LTE network. Both transport networks allow the CPE to reach the Internet. Details regarding the 5G network can be found in Table II.

The experiments were deployed over a 20 MHz LTE channel for the UMAHetNet and 50 MHz for the 5G pilot network. In both cases, no channel conditions were emulated because

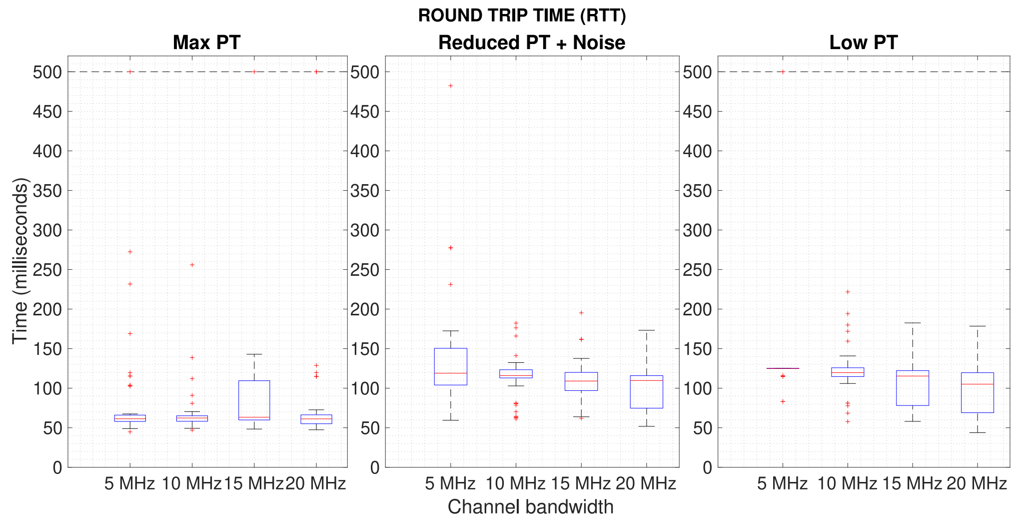


Fig. 15. Round trip time per channel scenario.

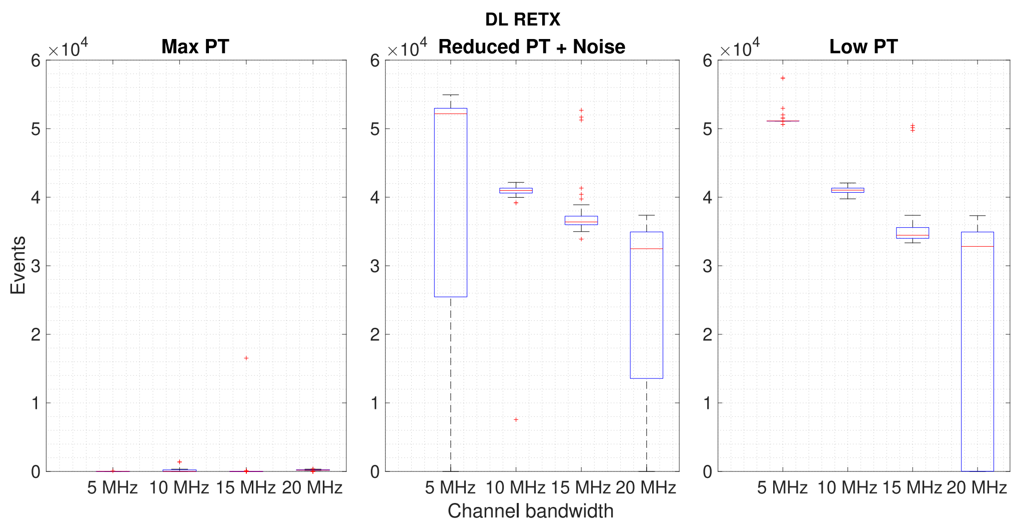


Fig. 16. Number of retransmissions per scenario.

TABLE II
5G NETWORK FEATURES

Parameter	Description
5G Mode	Standalone (SA)
Band	n78
Duplex mode	TDD
Carrier frequency	3774.990 MHz
Channel bandwidth	50 MHz
Max. TX power	1W
Antennas	4TX/4RX
Nominal gain	C0 dBi (Omnidirectional)
Beamforming	No

of the used real testing environment. Each test belongs to 120-second iterations. The complete experiment comprises 60 iterations for each technology assessed. The video client was not modified at all and the same configurations as presented in Table I are followed.

A statistical analysis of the initial playback time is presented in Figure 18. As it is possible to observe, the results obtained in 5G indicate a median time of 0.3s approximately, while LTE is around 0.4s. The key difference is that the initial times in 5G tend to be consistent along the experiment while LTE presents more dispersion around the median. Despite this fact, both technologies perform well, for this metric, in a similar fashion to the Crowdcell scenario with the best channel conditions.

Another indicator analyzed to quantify the quality of service is the *frame rate*. As can be seen in Figure 19, both LTE and 5G offer values close to 30 FPS on average. The CDFs refer that 5G has an approximate range of variation between 28 to 30 within its 25th and 75th percentiles, while LTE from 24 to 30. Nevertheless, both technologies are able to maintain a smooth 360-video service (because of displaying rate adaption managed by the player), in terms of frame rate. This is an important factor, as small changes in the sequence of images in VR can cause dizziness or confusion and consequent deterioration in the quality of user experience.

The main difference between the service offered by LTE with respect to 5G is the resolution at which the video can be played. Figure 20 shows how LTE uses about 65% of the

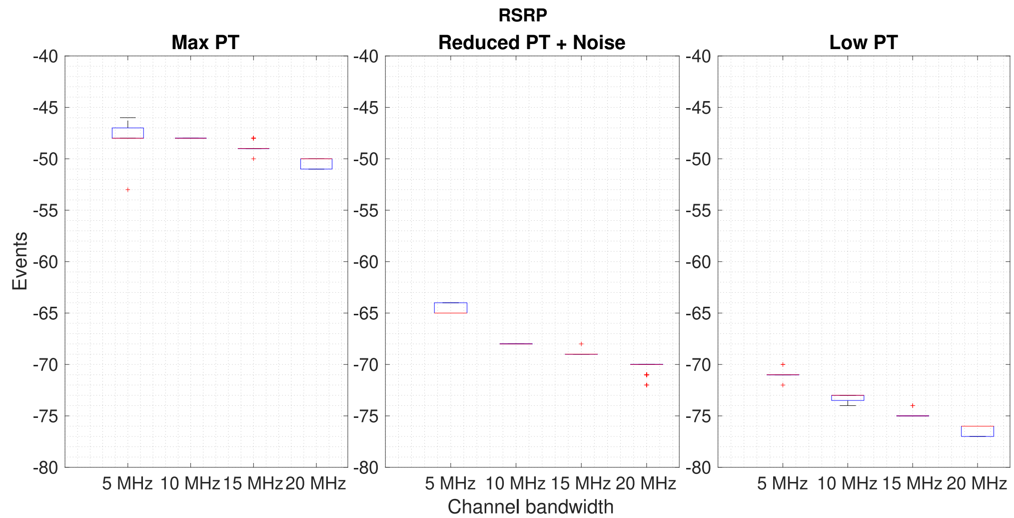


Fig. 17. RSRP measured for each experiment case.

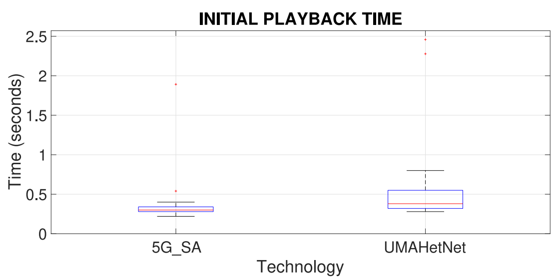


Fig. 18. Video initial playback time per technology.

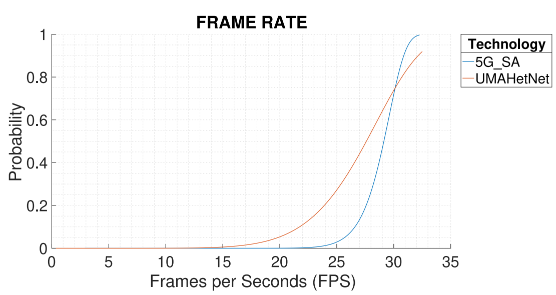


Fig. 19. Probability cumulative function per technology.

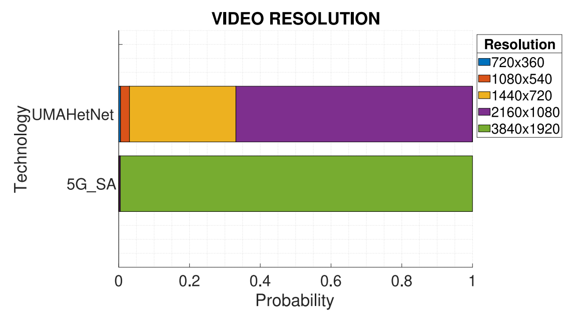


Fig. 20. Video resolution per technology.

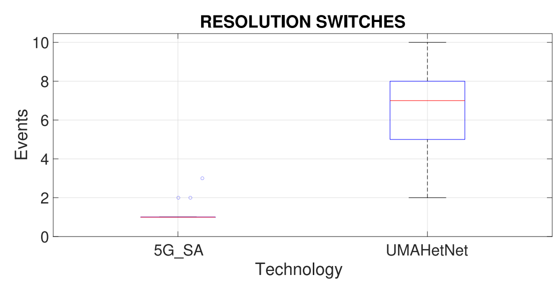


Fig. 21. Resolution switching events per technology.

samples the 2160x1080 resolution while the other half of the data reflects the use of lower quality resolutions. However, LTE tests show that this technology is not able to bear the maximum available resolution (3840x1920) for this media source. Conversely, the 5G scenario is different since about 99% of the samples show that this technology allows using the best possible resolution throughout the duration of the service.

Likewise, the number of resolution switches may improve the understanding of how each technology performs this service. As shown in Figure 21, 5G presents a very steady service, where most experiments keep the same resolution throughout the playback, which matches with the information displayed in Figure 20. In contrast, LTE displays at least 7 switches on average per session. This fact shows that commercial-based LTE networks could poorly perform causing a very unstable quality service with respect to the results obtained with the LTE Crowdcell.

Furthermore, Figure 22 presents a statistical analysis of the estimated throughput values per session, where LTE achieves a median value of 2.2 Mbps, while 5G achieves a median value of 5.5 Mbps. Although this difference seems to be insignificant, the values reached by 5G point out a similar performance in comparison with the results performed by the best case for LTE Crowdcell (in terms of resolution, frame rate, . . .). Nonetheless, it is remarkable to note that the current service conditions are not an actual challenge for 5G infrastructure. Indeed, the throughput measured only represents what this service needed to download the 360-video.

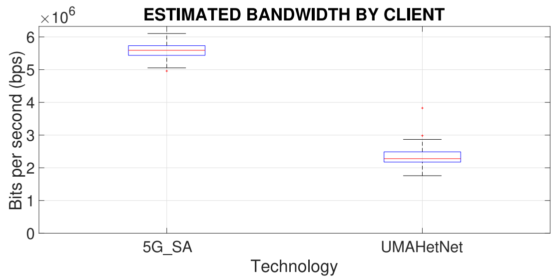


Fig. 22. Throughput per technology.

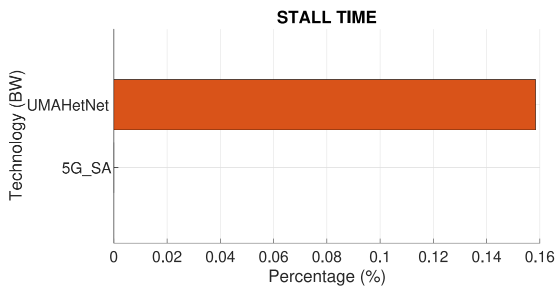


Fig. 23. Stall events per technology.

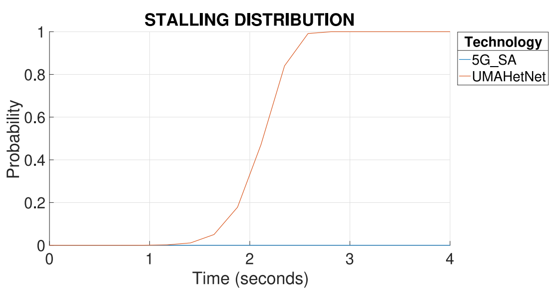


Fig. 24. Stall events distribution per technology.

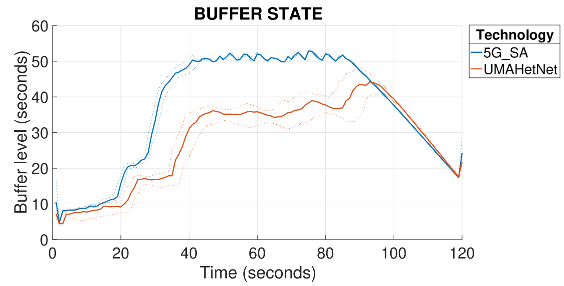


Fig. 25. Buffer state per technology.

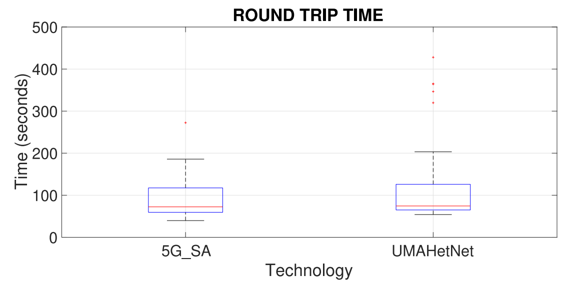


Fig. 26. Round trip time per technology.

In addition, it is possible to remark that the tested service did not present interruptions while the media was being played. Figure 23 shows that LTE performance is as stable as 5G in terms of stall or freezing events. However, the LTE QoE is intended not to be as high as 5G due to the nature of low and mid-resolution content.

Following this scope, Figure 24 depicts the temporal distribution of stalling events. As can be seen, the average stall in LTE rounds 2.1s, while 5G presents no stalling events along the experiments. A different perspective to understand these metrics is calculating the time that represents the total stalling time, which is approximately 11.52s for the 0.16%. Thus, it means that about 5 stalling events occurred during the testing.

Including this analysis, the buffer level state metric is presented in Figure 25. The results gathered suggest that the 5G network has enough resources to manage this service with no challenging issues. On the contrary, LTE shows a struggling performance intended to properly maintain the buffer level, avoiding possible considerable stalling events. Nonetheless, the resources are not enough to increase the video quality as shown by the resolution metrics.

To finish this analysis, the round trip time measurements are presented below. As depicted in Figure 26 5G experiments obtained an average latency of 72 ms while LTE 74 ms. Although these values seem to be comparable, it is important to notice some considerations. Firstly, LTE is bearing a mid-low quality service while 5G performs the best possible video quality. Second, these latency values come from commercial-based networks deployed in real scenarios, which increases the issues related to channel degradation in comparison with the scenario with LTE Crowdcell which serves an indoor-dedicated LTE cell.

In general terms, the experiments done in this testbed convey that VR applications, specially 360-video, require as wide downlink channel bandwidths as possible to guarantee enough throughput for high-quality video frames. LTE shows an affordable performance, for all the tested combinations (bandwidth, power and noise conditions). Despite those facts, 5G arises as the best choice in terms of wide channel bandwidth as well as reduced latency and network flexibility.

The results collected show that 5G and LTE Crowdcell best channel provides a suitable 360-video service. However, the resolution quality is still far from the recommended theoretical resolutions (at least 4k or even more per eye) due to hardware unavailability at this current time (close to 2k per eye and 120° FOV). The frame rate is not an actual issue because both technologies perform quite well, but 60 and 120 Hz may bring this service to a realistic real-feel scenario. In fact, the increase in resolution quality and frame rate will lead to very high throughputs that possibly will not be possible to manage with LTE technology.

Regarding the latency, the state-of-the-art values suggest a delay inferior to 20 ms for optimal immersion, which is currently impractical. However, the measured values display a close performance to the suggested values for good-quality service (< 60 ms) with 61 ms for LTE Crowdcell best channel and 72 ms for 5G in a real scenario. Based on this analysis, it may be possible to reduce this delay through optimization in the source by applying tiled-based strategies and low-latency streaming versions on protocols like DASH. In this same context, the results obtained in this work clearly show that network optimization strategies are needed to improve performance.

VI. Conclusion

Virtual reality has emerged as a technology of great interest to users, researchers, and content providers. In this context, the development of the new mobile technology, 5G, is expected to drive the massive use of services such as virtual reality due to the offered traffic volumes and reduced latencies. In addition, VR can take advantage of the benefits provided by cloud computing, thus reducing computational needs as well as bringing the content closer to the user, this way reducing the startup times and latencies. This all together with source optimization strategies such as tile-based content, may improve the visual quality while at the same time reducing the network stress.

The developed work has presented the design and implementation of a framework for the evaluation of VR, through its quality indicators KQIs for the case study of 360-video. This framework implements a client in the VR HMD to playback immersive multimedia DASH content, which is downloaded from a server located on the Internet.

This is applied to evaluate LTE and 5G networks communicating the client with the server. The measurements performed in this work have demonstrated the feasibility of achieving high-quality levels (resolution and frame rate) using mobile networks. On the one hand, the experiments describe a better service performance as the channel bandwidth is wide enough to transport high-quality video frames. The best LTE results were reached with 15 and 20 MHz channels.

On the other hand, additional experiments were done with changes in the radio channel conditions such as in the transmission power as well as in the noise level through the Crowdcell's SDR emulation tool. The results show that both factors severely degrade the service quality in combination with a constrained channel bandwidth, specifically 5 and 10 MHz. These are not recommended for 360-video service provision due to their limited and unstable video quality produced by low-resolution segment commutation and frame rate fluctuation.

Moreover, the testbed was assessed using a real deployed LTE picocell network (20 MHz) and a pilot Sub-6GHz 5G network (50 MHz). The 360-video service working over 5G exhibited improved performance in comparison with LTE. The video resolution and frame rate were stable enough and video frames used the highest possible resolution most of the time. In addition, the initial playback was clearly benefited by the wide channel bandwidth reaching an increased throughput. As expected, this condition assures constant traffic that prevents freezing events through the playback.

These results position 5G as a suitable technology to provide 360-video service, in addition to allowing the development of applications in environments that require mobility.

As a contribution, this work extends the possibilities of analysis that are being risen by new-generation network approaches such as mmWave radio, network slicing, SDN and edge computing (MEC) to meet the service requirements in terms of latencies lower than 20 ms and throughputs close to the 1 Gbps to assure real-feel VR experiences [19].

References

[1] F. Nielsen, "Surround video: A multihead camera approach," *Vis. Comput.*, vol. 21, no. 1, pp. 92–103, Feb. 2005. [Online]. Available: https://link.springer.com/article/10.1007/s00371-004-0273-z

[2] "YouTube VR—Home—YouTube VR." [Online]. Available: https://vr.youtube.com/ (Accessed: Sep. 7, 2021).

[3] C. L. Fan, W. C. Lo, Y. T. Pai, and C. H. Hsu, "A survey on 360 video streaming," *ACM Comput. Surveys*, vol. 52, no. 4, pp. 1–36, Aug. 2019. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3329119

[4] S. Rahimizhian, A. Ozturen, and M. Ilkan, "Emerging realm of 360-degree technology to promote tourism destination," *Technol. Soc.*, vol. 63, Nov. 2020, Art. no. 101411.

[5] F. Ulrich, N. H. Helms, U. P. Frandsen, and A. V. Rafn, "Learning effectiveness of 360° video: Experiences from a controlled experiment in healthcare education," *Interact. Learn. Environ.*, vol. 29, no. 1, pp. 98–111, 2021. [Online]. Available: https://www.tandfonline.com/action/journalInformation?journalCode=nile20

[6] R. C. Herault, A. Lincke, M. Milrad, E.-S. Forsgärde, and C. Elmqvist, "Using 360-degrees interactive videos in patient trauma treatment education: Design, development and evaluation aspects," *Smart Learn. Environ.*, vol. 5, no. 1, pp. 1–15, Dec. 2018. [Online]. Available: https://link.springer.com/articles/10.1186/s40561-018-0074-x

[7] H. T. Tran, N. P. Ngoc, C. T. Pham, Y. J. Jung, and T. C. Thang, "A subjective study on user perception aspects in virtual reality," *Appl. Sci.*, vol. 9, no. 16, p. 3384, Aug. 2019. [Online]. Available: www.mdpi.com/journal/applsci

[8] W. Shen, L. Ding, G. Zhai, Y. Cui, and Z. Gao, "A QoE-oriented saliency-aware approach for 360-degree video transmission," in *Proc. IEEE Int. Conf. Vis. Commun. Image Process. (VCIP)*, Dec. 2019, pp. 1–4.

[9] A. M. Gavgani, F. R. Walker, D. M. Hodgson, and E. Nalivaiko, "A comparative study of cybersickness during exposure to virtual reality and 'classic' motion sickness: Are they different?" *J. Appl. Physiol.*, vol. 125, no. 6, pp. 1670–1680, Dec. 2018. [Online]. Available: https://journals.physiology.org/doi/full/10.1152/japplphysiol.00338.2018

[10] E. Chang, H. T. Kim, and B. Yoo, "Virtual reality sickness: A review of causes and measurements," *Int. J. Human–Comput. Interact.*, vol. 36, no. 17, pp. 1658–1682, 2020. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/10447318.2020.1778351

[11] S. LaMotte. "Virtual reality has some very real health dangers—CNN." Dec. 2017. [Online]. Available: https://edition.cnn.com/2017/12/13/health/virtual-reality-vr-dangers-safety/index.html

[12] P. J. Costello, P. J. Costello, C. J. Rd, P. Costello, and L. Nl. "Health and safety issues associated with virtual reality—A review of current literature." 1997. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.6.3025

[13] Facebook Technologies. "Oculus quest 2 safety center—Oculus," 2022. [Online]. Available: https://www.oculus.com/safety-center/quest-2/

[14] P. Caserman, M. Martinussen, and S. Göbel, *Effects of End-to-End Latency on User Experience and Performance in Immersive Virtual Reality Applications* (LNCS, 11863 (Lecture Notes Artificial Intelligent Lecture Notes Bioinformatics)). E. van der Spek, S. Göbel, E. Y.-L. Do, E. Clua, and J. B. Hauge, Eds. Cham, Switzerland: Springer Int., Nov. 2019, pp. 57–69. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-34644-7_5

[15] J. DiZio and P. Lackner, "Motion sickness side effects and after-effects of immersive virtual environments created with helmet-mounted visual displays. NATO RTO-MP-54, the capability of virtual reality to meet military requirements," Defense Tech. Inf. Center, Fort Belvoir, VA, USA, Rep. ADA388966, 2000. [Online]. Available: https://apps.dtic.mil/dtic/tr/fulltext/u2/p010619.pdf

[16] M. Uitto and A. Heikkinen, "Exploiting and evaluating live 360° low latency video streaming using CMAF," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Jun. 2020, pp. 276–280.

[17] J. Yi, R. Islam, S. Aggarwal, D. Koutsonikolas, Y. C. Hu, and Z. Yan, "An analysis of delay in live 360° video streaming systems," in *Proc. 28th ACM Int. Conf. Multimedia*, 2020, pp. 982–990. [Online]. Available: https://doi.org/10.1145/3394171.3413539

[18] T. Taleb, Z. Nadir, H. Flinck, and J. Song, "Extremely interactive and low-latency services in 5G and beyond mobile systems," *IEEE Commun. Stand. Mag.*, vol. 5, no. 2, pp. 114–119, Jun. 2021.

[19] M. S. Elbamby, C. Perfecto, M. Bennis, and K. Doppler, "Toward low-latency and ultra-reliable virtual reality," *IEEE Netw.*, vol. 32, no. 2, pp. 78–84, Mar./Apr. 2018.

[20] E. Bastug, M. Bennis, M. Medard, and M. Debbah, "Toward interconnected virtual reality: Opportunities, challenges, and enablers," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 110–117, Jun. 2017.

[21] M. Dasari, A. Bhattacharya, S. Vargas, P. Sahu, A. Balasubramanian, and S. R. Das, "Streaming 360-degree videos using super-resolution," in *Proc. IEEE INFOCOM*, Jul. 2020, pp. 1977–1986.

[22] Y. Siriwardhana, P. Porambage, M. Liyanage, and M. Ylianttila, "A survey on mobile augmented reality with 5G mobile edge computing: Architectures, applications, and technical aspects," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 2, pp. 1160–1192, 2nd Quart., 2021.

[23] H. T. Tran, N. P. Ngoc, C. M. Bui, M. H. Pham, and T. C. Thang, "An evaluation of quality metrics for 360 videos," in *Proc. Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2017, pp. 7–11.

[24] P. Hanhart, Y. He, Y. Ye, J. Boyce, Z. Deng, and L. Xu, "360-degree video quality evaluation," in *Proc. Picture Coding Symp. (PCS)*, Sep. 2018, pp. 328–332.

[25] R. I. T. D. C. Filho *et al.*, "Dissecting the performance of VR video streaming through the VR-EXP experimentation platform," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 15, no. 4, pp. 1–23, Nov. 2019. [Online]. Available: https://dl.acm.org/doi/10.1145/3360286

[26] M. S. Anwar, J. Wang, A. Ullah, W. Khan, S. Ahmad, and Z. Fei, "Measuring quality of experience for 360-degree videos in virtual reality," *Sci. China Inf. Sci.*, vol. 63, no. 10, Oct. 2020, Art. no. 202301. [Online]. Available: https://link.springer.com/article/10.1007/s11432-019-2734-y

[27] S. Park, A. Bhattacharya, Z. Yang, M. Dasari, S. R. Das, and D. Samaras, "Advancing user quality of experience in 360-degree video streaming," in *Proc. IFIP Netw. Conf. (IFIP Netw.)*, May 2019, pp. 1–9. [Online]. Available: https://ieeexplore.ieee.org/document/8999460/

[28] A. Herrera-Garcia, S. Fortes, E. Baena, J. Mendoza, C. Baena, and R. Barco, "Modeling of key quality indicators for end-to-end network management: Preparing for 5G," *IEEE Veh. Technol. Mag.*, vol. 14, no. 4, pp. 76–84, Dec. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8868102/

[29] C. Baena, S. Fortes, E. Baena, and R. Barco, "Estimation of video streaming KQIs for radio access negotiation in network slicing scenarios," *IEEE Commun. Lett.*, vol. 24, no. 6, pp. 1304–1307, Jun. 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9031300/

[30] O. S. Peñaherrera-Pulla, C. Baena, S. Fortes, E. Baena, and R. Barco, "Measuring key quality indicators in cloud gaming: Framework and assessment over wireless networks," *Sensors*, vol. 21, no. 4, p. 1387, Feb. 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/4/1387

[31] B. Krogfoss, J. Duran, P. Perez, and J. Bouwen, "Quantifying the value of 5G and edge cloud on QoE for AR/VR," in *Proc. 12th Int. Conf. Qual. Multimedia Exp. (QoMEX)*, May 2020, pp. 1–4.

[32] C. Anthes, R. J. García-Hernández, M. Wiedemann, and D. Kranzlmüller, "State of the art of virtual reality technology," in *Proc. IEEE Aerosp. Conf. Process.*, Jun. 2016, pp. 1–19.

[33] S. S. Jung and J. Jeong, "A classification of virtual reality technology: Suitability of different VR devices and methods for research in tourism and events," in *Augmented Reality and Virtual Reality: Changing Realities in a Dynamic World*, T. Jung, M. C. tom Dieck, and P. A. Rauschnabel, Eds. Cham, Switzerland: Springer, 2020, pp. 323–332. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-37869-1_26

[34] "Cloud AR/VR streaming: Accelerate mass adoption and improve quality of experience of AR/VR using 5G and edge cloud," GSMA, London, U.K., Rep., Mar. 2019. [Online]. Available: https://www.gsma.com/futurenetworks/wp-content/uploads/2019/03/Cloud-ARVR-booklet-for-MWC19.pdf

[35] RenderHeads. "Releases RenderHeads/UnityPlugin-AVProVideo GitHub." May 2021. [Online]. Available: https://github.com/RenderHeads/UnityPlugin-AVProVideo/releases

[36] Google. "ExoPlayer." 2021. [Online]. Available: https://exoplayer.dev/

[37] *Study on Key Quality Indicators (KQIs) for Service Experience*, 3GPP Standard TS 32.862, 2015. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2260

[38] Bitmovin. "VR & 360. this demo shows the Bitmovin player displaying a 360° video with VR." 2022. [Online]. Available: https://bitmovin.com/demos/vr-360

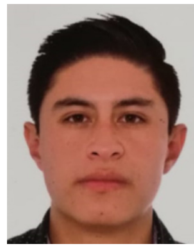
[39] ZeroTier. "ZeroTier—Global area networking." [Online]. Available: https://www.zerotier.com/ (Accessed: Oct. 4, 2021).

[40] C. Baena, S. Fortes, O. Penaherrera, and R. Barco, "A framework to boost the potential of network-in-a-box solutions," in *Proc. 12th Int. Conf. Netw. Future (NoF)*, 2021, pp. 1–3.

[41] Lime Microsystems. "LimeNET CrowdCell." 2020. [Online]. Available: https://limemicro.com/products/systems/limenet-crowdcell/

[42] S. F. Rodríguez, "Context-aware self-healing for small cell networks," Ph.D. dissertation, Dept. Comput. Sci., Universidad de Málaga, Málaga, Spain, 2017.

[43] S. Fortes *et al.*, "The campus as a smart city: University of Málaga environmental, learning, and research approaches," *Sensors*, vol. 19, no. 6, p. 1349, Mar. 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/6/1349



O. S. Peñaherrera-Pulla (Member, IEEE) received the engineering degree in electronics with a mention in telecommunications from the Universidad Politécnica Salesiana, Ecuador, in 2017, and the master's degree in telematics and telecommunication networks from the University of Málaga, Spain, in 2020, where he is currently pursuing the Ph.D. degree focused on gaming and AR/VR applications, their performance monitoring, and the assessment and optimization of these services for wireless networks. He is a Research Assistant with the University of Málaga.

Carlos Baena received the M.Sc. degree in telematic engineering from the University of Málaga, Spain, where he is currently pursuing the Ph.D. degree focused on E2E network optimization for video and gaming services and has been working as a Research Assistant with the Department of Communications Engineering since 2018.

Sergio Fortes (Member, IEEE) received the M.Sc. and Ph.D. degrees in telecommunication engineering from the University of Málaga, Spain. He began his career with the field of satellite communications, holding positions in European space agencies, where he participated in various research and consultant activities on broadband and aeronautical satellite communications. In 2012, he joined the University of Málaga, where his research is focused on the application of machine learning techniques for the advanced management of cellular, satellite, and IoT communications.

Eduardo Baena received the M.Sc. degree in telecommunication engineering from the Universidad de Granada, Spain, in 2010. He is currently pursuing the Ph.D. degree focus on unlicensed band applications and advanced network management schemes, University of Málaga, Spain. He has held various industry positions in several companies, including operators, service providers, and manufacturers. Since 2017, he has been working as a Lecturer and a Researcher with the Department of Communications Engineering, University of Málaga.

Raquel Barco received the M.Sc. and Ph.D. degrees in telecommunication engineering from the University of Málaga, Spain. In 2000, she joined the University of Málaga, where she is currently a Full Professor. She has worked in projects with major mobile communications operators and vendors and is an author of more than 100 high-impact papers.