Subjective Quality of Experience Assessment in Mobile Cloud Games

Henrique Souza Rossi[1], Niclas Ögren[2], Karan Mitra[1], Irina Cotanis[2], Christer Åhlund[1], Per Johansson[2]
[1]Mobile and Pervasive Computing, Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, Skellefteå, Sweden
[2]Infovista, Skellefteå, Sweden
Email: henrique.souza.rossi@ltu.se

***Abstract*—The rise of mobile cloud gaming (MCG) has necessitated understanding its impact on mobile network design and deployment for end users' QoE maximization. MCG is a dynamic service that requires stringent quality from network operators. Therefore, this paper investigates the subjective QoE of MCG over mobile networks played on smartphones. We conducted subjective tests (N=31); our results indicate that MCG is affected differently by QoS attributes such as packet loss (PL), round trip time (RTT) and jitter compared to cloud games and online mobile games. We identify that RTT values above 100 milliseconds significantly impact users' QoE, measured via the mean opinion score (MOS). Further, lower RTT values with high PL; and higher RTT values with low PL cause a strong negative effect on MOS. Lastly, bursty jitter seems to affect the MOS, while random jitter does not significantly impact MOS.**

***Index Terms*—Subjective tests, Quality of Experience, Quality of Service, mobile games, mobile networks**

I. Introduction

Cloud gaming (CG) refers to the execution of games and rendering of their graphics in the cloud, instead of traditional gaming hardware (e.g., PC, consoles and portable devices) [1]. The growth of this segment is reported to expand to 6 billion US dollars by the year 2024, and of which already includes an estimate of 23.7 million subscribers of CG services in 2021 [2]. As cloud environments support the deployment of high computational demanding real time applications on thin clients (e.g. a mobile phone), through network infrastructure, they allows games to be played efficiently on smartphones and create possibilities to access them anywhere and anytime [3].

A natural extension of CG is MCG which enables gaming-on-the-go [4]. This paper investigates the impact of mobile network degradation on users' perception of quality regarding MCG, or users' quality of experience (QoE) [5]. For that, we invited users (*N=31*) to play the first-person shooter (FPS) game called Counter-Strike: Global Offensive (CS:GO) streamed to a smartphone (through the usage of Steam Remote Play[1] while recording their QoE ratings (using 1 to 5 Likert-like scale (explained later) for 29 network scenarios (of which we computed the mean opinion score (MOS)). We considered network quality of service (QoS) factors such as PL, bursty jitter, random jitter, and RTT for network degradation. The major contributions of this paper are as follows:

1) To the best of our knowledge, this is the first paper to investigate subjective QoE regarding MCG from mobile network data perspective; and
2) We present the assessment of bursty and random jitter patterns on MCG.

This paper is organized as follows: Section II presents the related work. Section III presents our testbed implementation. Section IV presents subjective tests description. Section V presents the results.

II. Related Work

QoE for Cloud Gaming: Jarschel *et al.* [6] analyzed users' QoE in CG under various network conditions by studying the impact of QoS factors such as delay, throughput and PL based on commercial Sony PlayStation 3 (PS3) games. Clincy and Wilgor [7] carried out QoE evaluations for CG services under different network conditions regarding the FPS game, Borderland (using joystick and monitor). Lindström *et al.* [8] studied the impact of QoS factors affecting users' QoE and their in-game performance for both single player (Geometry Wars: Retro Evolved) and multiplayer (Speedrunners) game scenarios on clouds with varying combinations of PL and delay. Suznjevic *et al.* [9] conducted an analysis of Nivida GeForce Now service [2] adaption through a user study to assess player's QoE under different network conditions. Sabet *et al.* [10] proposed a latency compensation technique using game adaption that mitigates the influence of delay on QoE. Clincy and Wilgor [7] suggest that players of cloud-based FPS games are less tolerant to network latency and PL than players of other game genres; they also emphasized the differences between cloud vs non-cloud environments.

On the topic of MCG, Huang *et al.* [11] performed experiments with mobile devices running game hosted on the cloud with the focus on energy consumption, under different video factors (e.g., display resolution, bitrate and frame-rate). Wang and Dey [12], analyzed network and video factors e.g., frame rate, display resolution, and video codec and their affect on users' QoE for MCG. Its worth noting that the references mentioned above, do not investigate the effects of jitter on users' QoE, nor its consideration for MCG

[1]https://store.steampowered.com/remoteplay

[2]https://www.nvidia.com/en-us/geforce-now/

978-1-6654-3540-6/22

QoE for Gaming: Sabet *et al.* [13] investigated both subjectively and objectively the adaptability of gamer to frequently occurring delay patterns by playing modified open-source games (T-Rex, Shooting Range, Dodge and Ant Smasher). Liu *et al.* [14] studied the impact of delay between input device and the computer on which the user played CS:GO and how it effected their QoE. Laghari *et al.* [15], studied the effects of gaming QoE by considering both computer and mobile games, and factors such as video quality and the display size. Suznjevic *et al.* [16] investigated based on two user studies, the impact of network and system conditions, and game contexts on QoE using World of Warcraft. Dick *et al.* [17] analyzed the effects of delay, jitter and the users' gaming skills on their QoE in four multiplayer games (Counter Strike, Unreal Tournament, Need for Speed 2, and Warcraft 3). Schmidt *et al.* [18] investigated the importance of key network parameters (jitter, PL, delay and successive loss-probability) and its influence in QoE for online mobile gaming (OMG) (Fornite and PUBG on smartphones).

Although the aforementioned works did not target cloud environment, they shed light upon the network impact on QoE in games. Sabet *et al.* [13] indicate that frequent delay switching annoys gamers, which might lead to lower QoE. Dick *et al.* [17] suggests that latency and jitter have a game-dependent influence on QoE. Schmidt *et al.* [18] emphasized that network delay has more impact on QoE than PL.

Compared to the state-of-the-art research mentioned above, in this paper, we present our results based on comprehensive subjective tests conducted for mobile cloud gaming by bringing in the expertise from industry and academia. To the best of our knowledge, this paper presents the first work that investigates subjective QoE in mobile cloud games from mobile network data perspective.

III. Testbed

The goal of this research is to understand and establish the impacts of network degradation measured via quality of service (QoS) factors such as delay, jitter and PL on end user's QoE for mobile cloud gaming.

This paper aims to establish the impact of network degradation measured via quality of service (QoS) factors on end users' QoE for mobile cloud gaming. Game applications are well known to provide end users with rich multi-sensory experiences that include intensive visual graphics, high quality audio effects, support for a wide variety of input devices, gameplay mechanics, multiplayer options among other features that are accessible to developers by game development tools [19]. Claypool and Claypool [20], classified game actions regarding precision and deadline, and considered the FPS game actions among the most demanding in both attributes and highly sensitive to latency. As such, choosing a game within this genre is desirable for this study, considering network impairments could cause more significant disturbances during the user's gameplay. Among several FPS game titles, we selected CS:GO, due to its popularity as it is used for eSports competitions, supports the installation of dedicated servers (a vital requirement for the experiment) and has easy game match customization.

In this paper, we consider network jitter as the variation in delay. To emulate cloud gaming on the phone, our team developed a testbed comprising multiple computers that mimics the cloud environment in a lab setting. For that, two computers were used; named PC.A, a high-end gaming laptop that renders the game CS:GO and PC.B a high-end gaming laptop that runs the CS:GO dedicated server and controls bots, matches logic and rules. The former also acts as the streaming service, encoding the game frames and sending them over the network; this is illustrated in Figure 1. The streaming was done using Steam Remote Play.

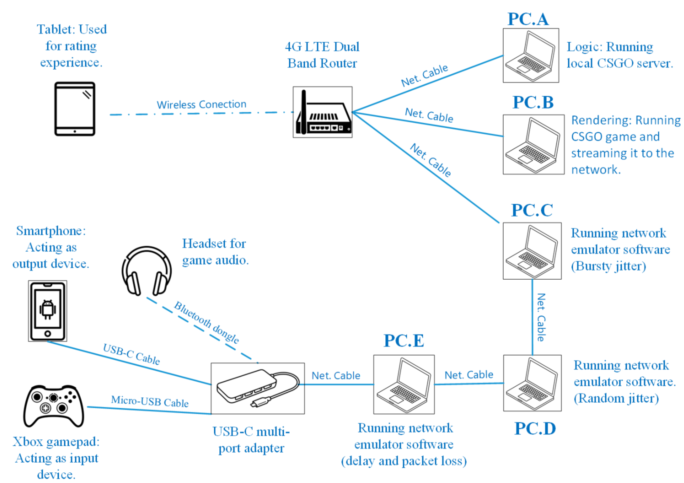

Fig. 1. Architecture diagram for the user test, showing the devices and their connection type.

The game is streamed from PC.A to the smartphone. However, in order to apply different network degradations, we added in between three computers: PC.C a Linux machine that emulates bursty Jitter (i.e applied in spikes following a certain interval); PC.D a Linux machine that emulates random jitter (i.e applied randomly throughout the game match); PC.E a Linux machine that emulates RTT and PL. We used Netem emulator [3] to emulate the network degradation parameters. Once the streaming packets goes through three computers, they arrive at the USB-C multi-port adapter (docking station) before reaching the smartphone for decoding.

To play the game, a gaming smartphone was used connected to a XBox One gamepad. The usage of a gamepad was due to the lack of touch support for CS:GO from Steam Remote Play. By using a gamepad's phone mount, the smartphone was naturally attached to the gamepad, making the scenario of playing on-the-go possible in this case. The streaming framerate on the phone was set at 120FPS. Hardware specifications used in our testbed is described in Table I.

a) CSGO Match Settings: The match the participants played for each network condition was customized in order to minimize the variability of gaming experience within the

[3]https://wiki.linuxfoundation.org/networking/netem

TABLE I
HARDWARE USED FOR CONDUCTING THE USER TEST.

Hardware	Device Specifications
PC.A	Ryzen 9 59000HX 3.3 GHz; Nvidia RTX 380; 16GB RAM; SSD; Win10
PC.B	Intel i7 8700 3.20Ghz; Nvidia RTX 2060; 32GB RAM; HDD; Ubuntu.
PC.C	Intel i7-4600 2.1 Ghz; HD Graphics 4400; 12GB RAM; HDD; Ubuntu
PC.D	Intel I7 4800U 2.7 Ghz; Intel Graphics 4400; 24GB RAM; HDD; Ubuntu.
PC.E	Inteli7-4710HQ 2.5 GHz; Nvidia GTX 860M; 8GB RAM; HDD; Ubuntu.
Router 4G LTE	TP-LinkAC1200 Dual Band Wi-Fi 4G LTE Router.
Tablet	Samsung Galaxy Tab S3 9.7
Smartphone - ROG Phone 5 Pro	Snapdragon 888 2.84 GHz; GPU Adreno 660; RAM 16GB. Android;

TABLE II
MATCH SETTINGS CUSTOMIZED AND APPLIED FOR ALL THE TESTS.

Setting	Description
Map Type	Dust 2
Match Goal	Kill Enemies
Weapon of Choice	Assault Rifle M4A4
Team of Choice	Counter-Terrorist
Adversary Type	Normal Bots
Number of Enemies	Max of 4 (enemies can respawn after death)
Round Duration	90s
Player Attribute1	Can respawn after death
Player Attribute2	Reduced damage by 80%

game match round and between users. For that, the choice of weapon, match map, objectives, number of enemies, and the match duration were kept the same for all tests. The full list of settings is described in Table II.

b) Network Conditions: The experiment network parameters and values presented in Table III emerged from three investigations:

Literature Review: The minimum and maximum values for the network attributes were identified. For RTT, the range of 0-400 milliseconds (ms) was considered [18]; for the PL, the range of 0-8% PL was considered [12]; for jitter the range of 0-150 ms was considered [17].

Real World Data Collection: This task was performed by recording real network data, by driving around the city of Skellefteå, Sweden. In particular, we recorded data using the TEMS [4] tool, by performing ICMP Echo 50 times per second (yielding 20ms between each measurement) towards the first responding hop, while moving or staying at a static position, at different hours of the day and targeting two LTE mobile networks. The results served us to better understand the behaviors of jitter and to distinguish between two types: i. Bursty jitter, which is characterized by delay spikes>=50 ms and possibility of frequency of occurrence between 0.01-1Hz. This frequency typically goes down when delay spikes increase. ii. Random jitter, which is frequent with small delay changes from packet to packet. Thus, a change in delay is always observed between two packets emulated from a normal distribution.

Pilot Test: We tested the initial parameters along with the system setup described in Figure 1, by running a pilot test. From the results, we refined the network conditions and compiled the final set of conditions used for the real experiment as described in Table III. For increasing the internal validity of the results, we followed the Latin Square Research design [21], such that each participant played all the network conditions, from a unique sequence order as a within-subject design.

[4]https://www.infovista.com/tems

TABLE III
NUMBER OF CONDITIONS AND SELECTED VALUES.

Parameter	N.Conditions	Values
RTT	6	25,50,100,200,300,400 in ms
PL	3	5%,25%,45% at RTT=2ms
PL and RTT	9	PL(0.2%,1%,5%); RTT(25,50,100)
Bursty Jitter	6	Jitter(50,200,1500 ms); Int(5s,15s,45s)
Noisy Rnd. Jitter	4	μ=25ms; Std(3,6,9,12)
Total	28	Final Count = 28 + 2xC0 [5]

IV. SUBJECTIVE TEST DESCRIPTION

User's QoE is a subjective metric that may be affected by several influencing factors pertaining to the usage of a system or service [5], [22]. For this research, the influencing factors (see Table III) caused network degradations applied in the MCG scenario. Users then provided their opinions regarding QoE through questionnaire. In our testbed, we included a tablet (see Figure 1), that displayed the questionnaire at the end of each round played by the users.

Each subjective test took on average 2 hours and 10 minutes. The flow of test was divided in the following steps: i. Read and sign consent form; ii. answering a demographics questionnaire and questions covering past experiences; iii. a warm up session was conducted for the users to get accustomed to the tests using the real test configuration but without any network conditions. Here, users were allowed to change the controller sensitivity in this phase; iv. conduct the real test where all the conditions were played in a unique randomized order by each user; and v. after the tests,feedback questions regarding the overall test experience were posed and answered.

a) User recruitment: We performed two experiments named the Pilot Test (Males=7) and the Real Test (Males=29, Females=2). For the Pilot Test, our team invited department colleagues who had prior experience playing games either on computers or smartphones. Regarding the Real Test, we advertised it among a student email list from university courses and technical programs. The requirement to participate was prior experience playing games on mobile phones or computers and being older than 18 years. For both tests, the users were presented with a text description of the evaluation procedure and signed a consent form.

b) Questionnaire: This research followed the demographics and game QoE questionnaire proposed by ITU study group 12 [23] with the following modifications: All the questions had the Likert-like scale of 1 to 5 (where '1' means "very poor", '2' means "poor", '3' means "fair", '4' means "good" and '5' means "excellent"). Another metric, 'System Acceptability', followed the original "yes"/"no" format. For the video quality metric, only one question was used: "how do you rate the overall video quality?". The demographics questionnaire had additional questions regarding the user's past experiences with network delay, for playing FPS games, and

[5]C0 = No degradation

TABLE IV
MEASUREMENT REPORT FOR MEAN (μ), STANDARD DEVIATION (SD), MEDIAM (M) AND INTERQUARTILE RANGE (IQR). JITTER = J; RTT = R

Condition	μ	SD	M	IQR	Condition	μ	SD	M	IQR
R=2ms,	4	0.7	4	0.5	R=50,PL=5%	1.7	0.8	2	1
R=100ms	3.5	0.9	4	1	R=100,PL=5%	1.3	0.5	1	1
R=200ms	2.3	1	2	1	J=25ms,STD=6	3.8	0.9	4	0.25
R=400ms	1.5	0.6	1	1	J=25ms,STD=9	2.2	0.7	2	0.5
R=2ms,PL=25%	3.6	0.8	4	1	J=200ms,Int=15s	3.4	3.5	1	1
R=2ms,PL=45%	1.1	0.2	1	0	J=1500ms,Int=15s	2.3	1	2	1
R=50,PL=0.2%	3.5	0.7	4	1	J=1500ms,Int=45s	2.2	1	2	2

CS:GO. The data was collected from 3rd of March till the 8th April 2022.

V. RESULT ANALYSIS

The results are illustrated in the Figure 2 divided in six graphs named (A)-(F), each with mean opinion score (MOS) (y-axis) is plot against the network attributes described in Table III. The MOS score reflects the mean of all the user ratings (using the 1 to 5 Likert-like scale mentioned above) for each test condition. The top of the bar shows the 95 % confidence interval, as a small vertical line, which was computed from a student's t-distribution. All tests conditions were not run equally many times across the population (N=31), due to a few technical issues while performing some of the user's tests. In this regard, the number of answers per conditions ranges between 28 to 31 (Mode (M)=31 and mean (μ)=30.1). The test conditions cited throughout this section, have their measurements reported in Table IV. The network PL, RTT and jitter values described in the graphs, refers to the system network conditions emulated by the NetEm tool on top of the network performance baseline of the hardware setup. They were applied both upstream and downstream. The lab measurements put the baseline RTT to be between 4-5ms 3ms, where the majority occurs on the link between the PC.E and the phone via the USB-C multiadapter. This baseline was not added to the RTT graph's measurement. The Figure 3 shows a boxplot for all tested cases and their respective QoE scores. The outliers (dots) follows the commonly used 1.5 * Interquatile range rule.

A. RTT Tests

The expectation from the RTT tests was to understand its impact on QoE; we also wanted to understand its interaction with PL. Our initial assumption was that higher RTT values cause a higher QoE impact on MCG since it delays the reception of feedback from performed input actions. In Figure 2 graph (A), MOS for seven conditions are illustrated. We see no apparent decrease in the MOS from RTT=2ms to RTT=100ms; the overlapping CI and a somewhat small IQR $<= 1$ indicate a small spread in the data, i.e. a more concise QoE opinion among the samples. This raises the concern of whether users can perceive small values of RTT for MCG or whether the presence of a gamepad or the size of the screen could have hindered their perception.

The best condition in the tests, where RTT=2ms, did not achieve a maximum score of 5. Although unexpected, it could be related to common bias from Likert scales such as avoidance of extreme responses [24] or system setup configuration could also have affected their baseline score. This effect was also noticed by Schmidt *et al.* [18] during the comparison of three other games (two CG, and one OMG). In our tests, the first sharp decrease perceived by users was seen when RTT was approx. 200ms, followed by a second decrease in MOS when the RTT was 400ms - leading to the lowest MOS in this group. Similar values were reported by Schmidt *et al.* for CS:GO played in CG, e.g. at RTT=400ms, it was around 2 for MOS and RTT=200ms being between 3 and 2. In contrast, for OMG (e.g., Fortnite), they reported considerable higher MOS across all delay tests (e.g. at RTT=400ms; it was between 4 and 3), which suggests a difference between OMG vs CG and MCG.

B. Packet Loss and RTT Tests

The tests regarding PL investigated two scenarios: PL with minor RTT (2ms) applied to NetEm described in Figure 2 graph (B), and PL interaction with different values for RTT described in Figure 2 graph (C). In both cases, the assumption is that the higher PL values lead to lower MOS. Graph (B) shows MOS for four PL conditions under RTT=2ms. There was no apparent change from PL=0% (the same RTT=2ms condition described previously) to PL=25% due to a slight MOS decrease and wide range from their CI. One possible cause for the minor distortions is that under the lower RTT, the Steam packet retransmissions could have been completed before the degradation effect being perceptible by users at least until 25%. From PL=25% to 45% there was a sharp and perceptual decrease in MOS. Schmidt *et al.* [18] reported MOS between 3 and 4 for bursty PL between 0% and 50%.

When it comes to the interaction of PL for higher values of RTT (25ms,50ms,100ms), graph (C) depicts MOS for 12 conditions (PL=0% conditions are shown in graph (A)). The data shows the the MOS decreases with an increase in PL values. For RTT=25ms, the changes in MOS are minor. We considered a perceptual impact on MOS for RTT=50ms group between PL=1% to 5% and for RTT=100ms for all the PL values (lowest at PL=5%) due to their small IQR and clear MOS changes. Our results differ significantly from Schmidt *et al.* [18] where at RTT=100ms and bursty PL=50% (10x more than ours) they reported MOS=3 for mobile gaming. It is unclear whether this difference is based solely on the burstiness of PL or the cloud environments (our scenario) that relies heavily on streaming video quality, which causes high video quality degradation during our internal tests for our PL cases.

C. Jitter Tests

In the jitter tests, the random jitter effect illustrated in Figure 2 graph (D), was applied constantly during the match with values coming from a normal distribution. The expectations was they would impact the sense of user input control (from gamepad) to the server. Graph (D) illustrates the MOS for four conditions that emulates the random jitter behavior from

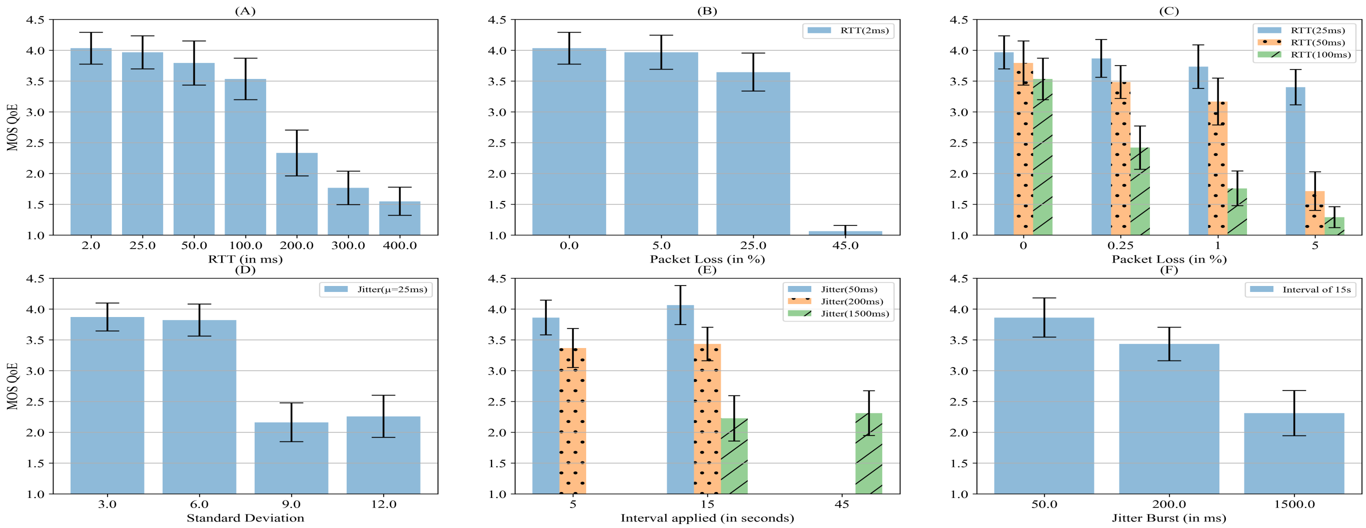

Fig. 2. Figures showing the impact of QoS factors such as round trip time delay (RTT), packet loss, and jitter on end user's QoE. These results are based on ratings gathered from approximately 30 participants for each use case.

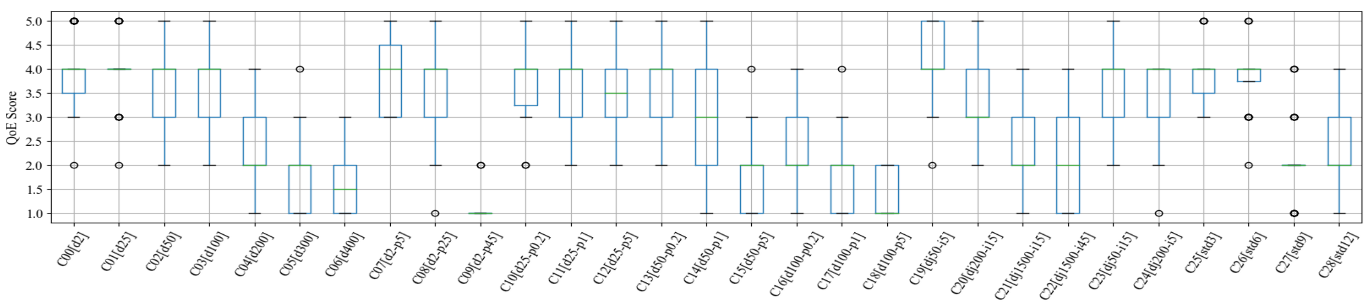

Fig. 3. Interquartile ranges for 29 use cases. Each one is represented as a tuple case numbers [qos factor-qos factor2]. For instance C15[d50-p5] means case 15 involving the RTT of 50 ms and PL of 5%. * d=RTT, i=intervals, p=packet loss, bj=burst jitter, std= standard deviation (random jitter)

a normal distribution. The distribution mean was kept at 25ms, and the standard deviation (STD) changed. Higher values for STD result in larger delay changes, it become more probable and thus more frequently appearing. The data shows a sharp decrease in MOS from STD=6 to STD=9, and very small IQR. During our internal tests a severe impact in video quality for STD=9 (a slight better quality for std=12) was seen, that might relate with users lower MOS scores. In addition, some users reported that although they had difficulties to see the game map, they were still able to easily control the game.

Regarding the tests for bursty jitter, the peaks between intervals of time that resembles a bursty pattern is covered in Figure 2 graph (E) for all of those cases. In addition, graph (F) shows a subset of cases with interval = 15s. Graph (E) details the MOS values when jitter was applied at different intervals (e.g. an interval of 5 means a delay burst value applied every 5s starting at CS:GO application execution). For this test, three peak values were selected (50ms, 200ms, and 1500ms) and applied with three different intervals (5s, 15s, and 45s). Our expectations was that smaller intervals at greater peaks, would cause more degradation to the network and negatively impact the MOS. From the peak size perspective in graph (F), the data shows a continuous degradation of MOS the bigger the peak value. This can be perceived at interval= 15s, when compared between the jitter=200ms, to the biggest jitter=1500ms. This variation suggests the bursty amplitude has an important role in MCG QoE.

On the other hand, from the interval side we evaluate there was no conclusive distortions in the MOS for all the peak groups. This data indicates minor to no change in QOE for MCG when it comes to the interval it is applied. Its noteworthy for Jitter=1500ms in the 45s interval, it had a large IQR showing an increased divergence of the users for this case. The work of Sabet *et al.*. [13] evaluated a similar effect regarding spikes, but for delay (with no jitter), and reporting the opposite effect of our results, as the MOS was impacted by more frequent spikes. It is unclear to us whether this differences are due to jitter effect, or due to the MCG environment. Thus, it could be further investigated.

On a general view from the network impairments effects on MOS, their scale on the video streaming and their perception by the users, they were influenced by the test setup and choice of game. CS:GO requires high control's precision from the players, and level of visual focus, for achieving game goals namely finding an enemy and kill. As such, games of different genres should entail different controlling requirements and impacting the MOS differently. A similar observation is made for the testing environment which was driven by steam link black-box streaming service. Different services might cause unforeseen input/output detriments to the QoE. Indeed, this can be perceived in [25], [26] on the differences found when benchmark various commercial cloud gaming services.

VI. Conclusion and Future Work

This paper contributes to the area of QoE by presenting a comprehensive analysis of the subjective tests conducted for MCG. Based on the data presented, the following findings can be highlighted:

- MOS for MCG decreases the most for RTT values higher than 100ms. For lower values results suggests no conclusive impact.
- The PL results indicates no apparent effect on MOS for values below 25 % under low RTT=2ms, but a strong negative effect for bigger values.
- For RTT=100ms under PL scenarios 0.2%,1%,5% it significantly affects the MOS for MCG, while smaller levels of RTT under the same PL scenarios had no conclusive effect.
- Bursty jitter pattern did not affect MOS based on the interval it was set, but instead show a clear negative effect regarding the size of the peak (greater the peak the lower the MOS).
- Regarding the random jitter pattern there was no apparent effect on MOS for values generated below STD<=6, but a considerable outcome for STD above 6.
- The-state-of-art reports for OMG, described MOS values with possible differences from our work for PL, RTT and bursty jitter when comparing to MCG. Further study is thus needed to verity the strength of those differences and its significance.

In the future, we aim to construct a gaming QoE model using the QoS metrics as input and outputs an objective estimate of the average QoE for cloud gaming over mobile networks, and partake in ITU-T standardization activities around measuring mobile network provided internet access with regards to cloud gaming.

VII. Acknowledgment

We would like to thank Solomon Sunday Oyelere and Ashmita Thapa for their help in conducting the subjective data collection, and to all the users who participated of our research.

References

[1] 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, publisher: MDPI AG.

[2] R. Newzoo, "Global Cloud Gaming Report The Infrastructure Edition," Tech. Rep., 2021.

[3] W. Cai, R. Shea, C.-Y. Huang, K.-T. Chen, J. Liu, V. C. M. Leung, and C.-H. Hsu, "A Survey on Cloud Gaming: Future of Computer Games," *IEEE Access*, vol. 4, pp. 7605–7620, 2016.

[4] S. Engl and L. E. Nacke, "Contextual influences on mobile player experience – A game user experience model," *Entertainment Computing*, vol. 4, no. 1, pp. 83–91, Feb. 2013.

[5] K. Mitra, A. Zaslavsky, and C. Ahlund, "Context-Aware QoE Modelling, Measurement, and Prediction in Mobile Computing Systems," *IEEE Transactions on Mobile Computing*, vol. 14, no. 5, pp. 920–936, May 2015.

[6] M. Jarschel, D. Schlosser, S. Scheuring, and T. Hoßfeld, "An Evaluation of QoE in Cloud Gaming Based on Subjective Tests," in *2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, Jun. 2011, pp. 330–335.

[7] V. Clincy and B. Wilgor, "Subjective Evaluation of Latency and Packet Loss in a Cloud-Based Game," in *ITNG '13*, Apr. 2013, pp. 473–476.

[8] S. Flinck Lindström, M. Wetterberg, and N. Carlsson, "Cloud Gaming: A QoE Study of Fast-paced Single-player and Multiplayer Gaming," in *IEEE/ACM UCC '20*, 2020, pp. 34–45.

[9] M. Suznjevic, I. Slivar, and L. Skorin-Kapov, "Analysis and QoE evaluation of cloud gaming service adaptation under different network conditions," in *QoMEX '16*, 2016, pp. 1–6.

[10] S. S. Sabet, S. Schmidt, S. Zadtootaghaj, B. Naderi, C. Griwodz, and S. Möller, "A latency compensation technique based on game characteristics to mitigate the influence of delay on cloud gaming quality of experience," in *MMSys '20*. ACM, 2020, pp. 15–25.

[11] C.-Y. Huang, Y.-L. Huang, Y.-H. Chi, K.-T. Chen, and C.-H. Hsu, "To Cloud or Not to Cloud: Measuring the Performance of Mobile Gaming," in *2nd Workshop on Mobile Gaming*. ACM, 2015, pp. 19–24.

[12] S. Wang and S. Dey, "Modeling and Characterizing User Experience in a Cloud Server Based Mobile Gaming Approach," in *GLOBECOM IEEE Global Telecommunications Conference*, 2009, pp. 1–7.

[13] S. S. Sabet, S. Schmidt, C. Griwodz, and S. Möller, "Towards the Impact of Gamers' Adaptation to Delay Variation on Gaming Quality of Experience," in *QoMEX '19*, Jun. 2019, pp. 1–6, iSSN: 2472-7814.

[14] S. Liu, M. Claypool, A. Kuwahara, J. Sherman, and J. J. Scovell, "Lower is Better? The Effects of Local Latencies on Competitive First-Person Shooter Game Players," in *CHI 21'*. ACM, 2021, pp. 1–12.

[15] A. A. Laghari, K.-u.-R. Laghari, K. A. Memon, M. B. Soomro, R. A. Laghari, and V. Kumar, "Quality of experience (QoE) assessment of games on workstations and mobile," *Entertainment Computing*, vol. 34, p. 100362, May 2020.

[16] M. Suznjevic, L. Skorin-Kapov, A. Cerekovic, and M. Matijasevic, "How to measure and model QoE for networked games?" *Multimedia Systems*, vol. 25, no. 4, pp. 395–420, 2019.

[17] M. Dick, O. Wellnitz, and L. Wolf, "Analysis of factors affecting players' performance and perception in multiplayer games," in *ACM SIGCOMM - NetGames '05*. ACM Press, 2005, p. 1.

[18] S. Schmidt, S. Zadtootaghaj, S. S. Sabet, and S. Möller, "Modeling and Understanding the Quality of Experience of Online Mobile Gaming Services," in *QoMEX '21*, Jun. 2021, pp. 157–162, iSSN: 2472-7814.

[19] E. Christopoulou and S. Xinogalos, "Overview and Comparative Analysis of Game Engines for Desktop and Mobile Devices," *International Journal of Serious Games*, vol. 4, no. 4, Dec. 2017.

[20] M. Claypool and K. Claypool, "Latency can kill: Precision and deadline in online games," in *MMSys '10*. ACM Press, 2010, pp. 215–222.

[21] D. C. Montgomery, *Design and analysis of experiments*, eighth edition ed. Hoboken, NJ: John Wiley & Sons, Inc, 2013.

[22] ITU-T Recommendation P.10/G.100, "Vocabulary for performance, quality of service and quality of experience," Tech. Rep., Nov. 2017.

[23] ITU-T Contribution SG12.C293, "Data Assessment for an Opinion Model Predicting Gaming QoE (G.OMG)," Tech. Rep., 2018.

[24] A. Furnham, "Response bias, social desirability and dissimulation," no. Personality and individual differences 7.3, pp. 385–400, 1986.

[25] F. Aumont, F. Humbert, C. Neumann, C. Salmon-Legagneur, and C. Taibi, "Dissecting cloud game streaming platforms regarding the impacts of video encoding and networking constraints on QoE," in *Proceedings of GameSys '21*. ACM, 2021, pp. 13–19.

[26] A. Di Domenico, G. Perna, M. Trevisan, L. Vassio, and D. Giordano, "A Network Analysis on Cloud Gaming: Stadia, GeForce Now and PSNow," *Network*, vol. 1, no. 3, pp. 247–260, 2021, publisher: MDPI AG.