QUICker or not? - An empirical analysis of QUIC vs TCP for video streaming QoE provisioning

Michael Seufert, Raimund Schatz, Nikolas Wehner, Pedro Casas

Angaben zur Veröffentlichung / Publication details:

Seufert, Michael, Raimund Schatz, Nikolas Wehner, and Pedro Casas. 2019. "QUICker or not? - An empirical analysis of QUIC vs TCP for video streaming QoE provisioning." In *22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN), 19-21 February 2019, Paris, France*, edited by Alex Galis, Fabrice Guillemin, Filip Idzikowski, and Muge Sayit, 7–12. Piscataway, NJ: IEEE. https://doi.org/10.1109/icin.2019.8685913.

QUICker or not? - an Empirical Analysis of QUIC vs TCP for Video Streaming QoE Provisioning

Michael Seufert, Raimund Schatz, Nikolas Wehner, Pedro Casas
AIT Austrian Institute of Technology GmbH, Vienna, Austria
michael.seufert.fl@ait.ac.at, raimund.schatz@ait.ac.at, nikolas.wehner@ait.ac.at, pedro.casas@ait.ac.at

***Abstract*—The introduction of the QUIC (Quick UDP Internet Connections) transport protocol by Google aimed to improve the Quality of Experience (QoE) with web services compared to the prevailing Transport Control Protocol (TCP). Nowadays, QUIC has become the default protocol to communicate between the Google Chrome browser and Google servers and accounts for an increasing share of the Internet traffic. This work investigates whether the promised QoE benefits of QUIC are indeed noticeable for end users or not. A measurement study was conducted for YouTube video streaming in two mobile and two fixed access networks, in which a defined set of videos was streamed back-to-back with QUIC and TCP in randomized order. QoE factors of video streaming (such as initial delay, the visual quality of the video, and stalling) were compared statistically to find significant differences between the streaming over QUIC and the streaming over TCP. Surprisingly, no evidence for any QoE improvement of QUIC over TCP in the context of YouTube streaming could be found.**

I. Introduction

Since its introduction in 1974, the TCP/IP suite with the transport-layer protocol TCP (Transmission Control Protocol) [1] has been by far the most important concept of today's Internet. While the underlying link layer saw and still sees the advent of many network technologies, such as today's fiber optics and 5G mobile networks, applications have factually converged to (encrypted) traffic over TCP. TCP provides reliable, ordered, and error-checked delivery of data and additionally implements flow control and congestion control mechanisms [2]. However, modern web applications might experience problems on the transport layer, which could not have been foreseen when TCP was designed. These include the high connection setup latency of three round trip times, head of line blocking, or bandwidth fairness of competing flows. Many different TCP versions have been proposed to overcome some of the problems, but most of these versions did not become practically significant, such that the original problems of TCP still prevail today [3], [4].

In 2013, Google presented the transport protocol QUIC (Quick UDP Internet Connections) [5], which was designed to reduce the connection and transport latencies inevitable in the prevailing TCP protocol. It is based on UDP (User Datagram Protocol), but adds a cryptographic handshake to allow for zero round trip time (RTT) connection setup to known servers and connection migration. Moreover, it implements loss recovery over UDP, supports multiplexed connections without head-of-line blocking, and moves congestion control to the application and the user space, which enables a rapid evolution for the protocol, as opposed to kernel space TCP [6].

Today, QUIC is still an experimental protocol, which is currently supported only by few browsers and servers. Moreover, firewalls are often configured to drop UDP traffic, which will also effectively block all QUIC traffic. However, QUIC has already become an integral part of Google services [7], as it is the default protocol for communication between Google Chrome browsers and Google servers. According to data gathered from 30 mobile operators [6], QUIC accounted for 20% of the total mobile traffic in November 2017, while 64% of the total QUIC traffic was video streaming traffic. QUIC's share is expected to further grow to 32% by November 2018. According to traffic measurements from a backbone link and a Tier-1 ISP, which were collected in 2017 and analyzed in [8], QUIC traffic only accounted for less than 10% of Internet traffic. This share was dominated by Google, which served up to 42.1% of its traffic via QUIC. Recently, it was announced that HTTP over QUIC will become HTTP/3, the upcoming major version of the HTTP protocol [9]. Thus, the share of QUIC will increase, and research on Quality of Experience (QoE) management and optimization has to consider it.

The eventual goal of the introduction of QUIC was to achieve a higher QoE for customers of Google services, typically achieved by faster page load times [10] for web browsing based services like Google Search or Google Docs, and less stalling and a higher visual quality [11] for video streaming service YouTube. These services are used by billions of users every day, and it was shown that a reduced QoE results in a reduction of service revenues [12]. Google announced first results [7], [13] that QUIC makes page loading 5% faster on average and 1 s faster for web search at the 99 percentile. Moreover, users reported 30% fewer stalling events for YouTube video streaming [13]. According to these results, the usage of the QUIC protocol could already be considered an effective form of QoE management.

This work investigates whether the usage of QUIC really yields noticeable benefits to end users of Google's video streaming service YouTube compared to the usage of TCP. A measurement study was conducted in two mobile and two fixed access networks from the perspective of a naive end user, who just uses his private Internet access to stream and consume a YouTube video. Using an automated video QoE measurement framework, the same videos were streamed back-to-back with QUIC and TCP in randomized order. Perceivable QoE factors were monitored on application level [11], such as initial delay, the visual quality of the video, and stalling. The QoE factors were compared statistically to find significant differences between streaming over QUIC versus streaming over TCP.

The remainder of the paper is structured as follows. Section II describes related works on QoE of video streaming and performance studies for QUIC. Section III introduces the measurement setup and the study design. The comparison of QoE factors for QUIC and TCP is conducted in Section IV, and Section V concludes.

II. Related Work

Most works on QoE of (adaptive) video streaming agree that initial delay, stalling, and quality adaptation are the most dominant QoE factors [11]. Among these, stalling, i.e., the playback interruptions due to buffer depletion, is considered the worst QoE degradation [14]. Moreover, it is important to reach a high played out video quality [15], while the initial delay has only a small impact on the QoE [16].

Apart from Google's report that QUIC achieved 30% fewer stalling events for YouTube video streaming [13], few works have considered the impact of QUIC on video QoE. [17] measured page load times of the YouTube website and found that QUIC outperforms TCP in unstable networks such as wireless mobile networks, but no obvious benefits could be found for stable and reliable networks. [18] extended previous works, which mainly focused on page load time experiments, and investigated video streaming performance in controlled environments over high-speed links with small packet loss. They found that QUIC can outperform TCP for video streaming only for high resolutions. [19] found that QUIC achieved shorter initial delays for YouTube than TCP especially with increasing RTT or packet loss, but only when leveraging its zero RTT connection setup. [20] compared adaptive video streaming over QUIC and TCP in testbed and Internet measurements. Here, QUIC could not outperform TCP, but instead resulted in a lower streamed video bitrate. In similar testbed experiments, [21] found that only QUIC with a higher number of emulated connections than default could reach a higher video streaming QoE than TCP.

These partially contradicting results motivate this work, which analyzes QUIC vs TCP for video streaming QoE provisioning from an end user perspective.

III. Methodology

Over a period of several days in autumn 2018, 916 YouTube video sessions were streamed and recorded on a measurement laptop in a home setup. For this, a Java-based monitoring tool similar to [22] was used. It used the Selenium browser automation library to automatically start a Chrome browser and browse to a single random YouTube video and stream for 180 s. The chrome browser was configured such that all HTTP requests were logged to a file (`-log-net-log`). A JavaScript-based monitoring script [23] was injected into the web page to record every 250 ms the current timestamp, as well as the current video playtime, buffered playtime, video resolution, and player state. This application-layer information about the streaming session was also logged to a file.

TABLE I: Maximum speed in considered networks measured by online speedtests.

network	downlink [Mbps]	uplink [Mbps]
M1	22.3	7.6
M2	10.1	6.0
F1	18.4	3.4
F2	13.0	0.9

458 random YouTube video IDs were generated. Each video was streamed twice, once with QUIC traffic enabled in the Chrome browser (`--enable-quic`), and once with QUIC traffic disabled, i.e., the video was streamed over TCP. Both video sessions were measured back-to-back, i.e., directly one after the other with only a short break of 1 min between the sessions. This should ensure that both streaming sessions should face similar network conditions, although this influence factor was not controlled in the measurement setup. For each video, the order of the streaming sessions (i.e., first QUIC session, second TCP session, or vice versa) was randomized and the browser cache was cleared after each session to avoid any effect of the serial position.

The video sessions were streamed in four different access networks to obtain more generalizable results, i.e., in a mobile network in Austria (M1), roaming in a mobile network in Italy using the same Austrian SIM card (M2), a fixed home network in Austria (F1), and a fixed home network in Italy (F2). For the measurements in the mobile networks, the measurement laptop was connected via Wi-Fi to a NetGear AirCard 785 Mobile Hotspot, which established the Internet connection over LTE using the Austrian SIM card. For the measurements in the fixed networks, the measurement laptop was directly connected to the home router via Wi-Fi. Table I indicates the maximum down- and uplink speeds of the four networks as observed with dedicated online speedtests, which shows that the networks offered largely sufficient bandwidth for the measurement client. Note again that the network was not controlled in this study and could be subject to bandwidth fluctuations or congestion. Only for some videos, a bandwidth limitation to 1 Mbps on both down- and uplink was applied using the tool NetLimiter 4 on the measurement laptop. The other videos were streamed without any bandwidth limitation (unlimited). Note that QUIC and TCP sessions belonging to the same YouTube video ID were streamed using the same conditions in terms of network and bandwidth limitation.

During the whole streaming session, the network traffic was captured using tshark. In each network trace, YouTube video flows were identified based on the domain name (googlevideo.com). Inspecting these flows made sure that the videos were either streamed via TCP or QUIC, just as configured by the measurement application. Moreover, the application-layer information about the streaming session were inspected to ensure that both the corresponding QUIC and TCP sessions did not contain an advertisement clip.

The final data set consisted of 504 streaming sessions

TABLE II: Measured streaming sessions per network and bandwidth limitation.

network	unlimited	1 Mbps
M1	58	52
M2	42	62
F1	62	50
F2	88	90

in a factorial design with three independent variables, i.e., protocol (QUIC/TCP), network (M1/M2/F1/F2), and limitation (unlimited/1 Mbps). This resulted in 31.5 video sessions per combination of independent variables on average, with a minimum of 21 video sessions per combination. Table II shows the numbers of streaming sessions per combination of network and bandwidth limitation in detail. Note again that each combination of network and limitation contains a set of QUIC sessions and a set of TCP sessions, which are of equal size and contain the same videos. Thus, this factorial design is especially suited to compare the performance of streaming over QUIC to streaming over TCP.

Six QoE factors of video streaming were considered as dependent variables, namely, initial delay, number of quality changes, average video resolution, average bitrate, number of stalling events, and total stalling time. These QoE factors were computed from the application-layer information logged for every streaming session.

IV. Evaluation of Impact on QoE Factors

The impact of the independent variables (protocol, network, limitation) on the six QoE factors is evaluated based on two statistical methods. First, the Kolmogorov-Smirnov (KS) test is applied, which decides on the null hypothesis that two samples were obtained from the same distribution, i.e., they cannot be distinguished statistically, based on the maximum vertical distance between the cumulative distribution functions (CDFs). If streaming over QUIC affected the streaming sessions differently than streaming over TCP, i.e., it would result in different QoE factors, the resulting distributions of the measured samples of the QoE factors would be different, and the null hypothesis would have to be rejected.

As a second method, Analysis of Variance (ANOVA) is conducted to identify differences between groups of measured samples based on the variation within and between different groups. In a post-hoc analysis based on Tukey's Honestly Significant Difference (HSD) test, the means of the groups are compared. If QUIC showed a significant performance improvement over TCP, the QoE factors of the QUIC sessions would have significantly "better" means compared to those of the corresponding TCP sessions. Here, "better" means lower in terms of initial delay, number of quality changes, number of stalling events, and total stalling time, but higher in terms of average resolution and average bitrate.

A. *Kolmogorov-Smirnov Test*

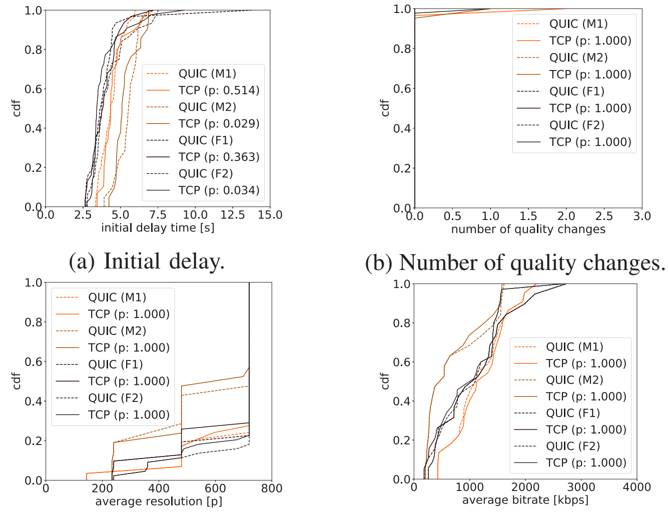
As each combination of network and limitation contains two sets of exactly the same videos, one with videos streamed over QUIC and one with the same videos streamed over TCP, the distributions of the QoE factors of these corresponding sets can be compared. Figure 1 shows these compared cumulative distribution functions (CDFs) of the initial delay, the number of quality changes, the average video resolution, and the average bitrate in the case without any bandwidth limitation (unlimited). Note that in this condition, no stalling occurred. Therefore, the QoE factors number of stalling events and total stalling time are omitted. Each plot in Figure 1 shows eight CDFs. The different networks are distinguished by color from orange (M1), light brown (M2), dark brown (F1) to black (F2). For each network, the dashed line depicts the CDF of the QUIC sessions, and the solid line is the CDF of TCP sessions.

Figure 1a shows the CDFs for the initial delays, i.e., the time from the start of the browsing until the playback start of the video, which are at least 2.5 s for all conditions, but can range up to 13.6 s in case of QUIC (F1). It can be seen that the CDFs show a very similar shape. A two-sample Kolmogorov-Smirnov (KS) test was conducted on the null hypothesis that the QUIC initial delay samples and the corresponding TCP initial delay samples were obtained from the same continuous distribution. The p-values of KS test are 0.514 for M1, 0.029 for M2, 0.363 for F1, and 0.034 for F2, as indicated in the legend for each of the four networks. The p-values for M1 and F1 are high, in particular higher than a typical significance level of 5% (0.05), which means that the hypotheses that the distributions of the corresponding QUIC and TCP samples are the same cannot be rejected. However, a significant difference was detected for M2 and F2. Inspecting the corresponding CDFs, M2 in light brown and F2 in black, the distributions of QUIC are located slightly more towards higher initial delay times, which indicates that QUIC actually performs slightly worse than TCP for most of the sessions in these networks. Nevertheless, the horizontal shift between the QUIC and TCP CDFs is at most 0.5 s and according to the results in [16], this slightly larger initial delay should have no impact on the resulting video streaming QoE.

Figure 1b shows the CDFs for the number of quality changes. It can be seen that the measured sessions contain almost no quality changes. Only very few TCP sessions (less than 5%) show one or two quality changes. As all p-values are 1, the KS tests indicate that the corresponding QUIC and TCP CDFs cannot be considered as being significantly different.

Figure 1c investigates the average video resolution of the video streaming sessions. Several plateaus can be observed in the CDFs, which correspond to the different YouTube quality levels at 240p, 360p, 480p, and 720p. While all CDFs are very similar, for all networks, more QUIC sessions can be streamed in the highest resolution of 720p than TCP sessions. However, the KS test and its maximum p-values of 1 again do not indicate any differences between the two protocols. A similar observation can be made for the average bitrate of the streaming sessions as depicted in Figure 1d. Again, all KS tests cannot reject the null hypothesis that the QUIC and TCP samples were obtained from the same distribution.

Figure 2 holds the corresponding results for the bandwidth limitation of 1 Mbps. Figure 2a investigates the initial delay.



(a) Initial delay.

(b) Number of quality changes.

(c) Average resolution.

(d) Average bitrate.

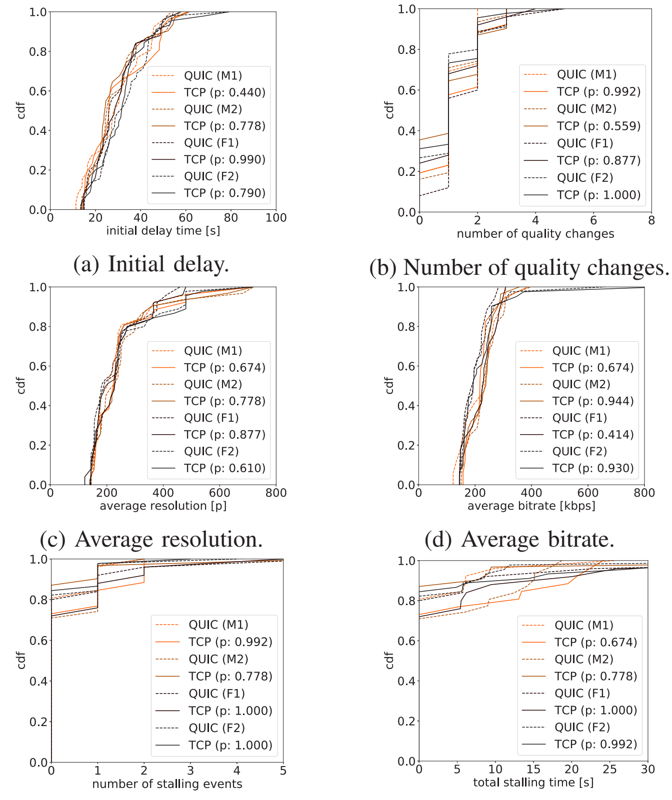
Fig. 1: QoE metrics for different networks and protocols in case of no limitation. Note that no stalling occurred.

All CDFs show a very similar shape starting from around 14 s and increase up to more than 50 s. Only some QUIC sessions in M1 show a smaller initial delay below 14 s. However, when looking at the p-values of the KS test, again the null hypothesis cannot be rejected, which indicates that there is no significant difference between the two protocols.

In contrast to the unlimited condition, the sessions streamed with a bandwidth limitation of 1 Mbps face a considerable number of quality changes as depicted in Figure 2b. Again, the differences between QUIC and TCP sessions are not significant, which is confirmed when investigating the average video resolution in Figure 2c or the average bitrate in Figure 2d. Both figures show that the bandwidth limitation decreases the average video resolution, even down to the lowest YouTube quality level, and accordingly the average bitrate, which is closely linked to the video resolution. Again, the CDFs show a highly similar behavior, which is confirmed by the high p-values of the KS tests.

Figure 2e shows that with a bandwidth limitation of 1 Mbps stalling could not be completely avoided. The ratio of streaming sessions with stalling ranges from 13% for TCP (M2) up to 29% for QUIC (M2). Figure 2f shows the corresponding CDFs for the total stalling time. It can be seen that stalling of more than 35 s could occur, which is a considerable amount for a total streaming duration of 180 s. The KS tests for both the number of stalling events and the total stalling time result in high p-values, such that the distributions cannot be considered as significantly different.

All in all, except for the initial delays in two networks without bandwidth limitation, the results of all Kolmogorov-Smirnov tests show that the null hypothesis (the measured samples of QoE factors from QUIC sessions and TCP sessions are from the same distribution) cannot be rejected. The significant differences for the initial delay indicate that streaming over QUIC might lead to a higher initial delay,



(a) Initial delay.

(b) Number of quality changes.

(c) Average resolution.

(d) Average bitrate.

(e) Number of stalling events.

(f) Total stalling time.

Fig. 2: QoE metrics for different networks and protocols in case of limitation to 1 Mbps. Here, stalling occured.

which contradicts its design goals. However, the differences are very small, and should have a negligible influence on the QoE [16]. Thus, in the measured video sessions there is no strong indication that streaming over QUIC results in a different QoE than streaming over TCP.

B. Analysis of Variance

To complement the results of the KS test, the impact of the independent variables on the QoE factors is investigated using multi-factor analysis of variance (ANOVA). A three-factor ANOVA was conducted for each QoE factor, to analyze the individual and combined influence factors of the three independent variables, namely, protocol (QUIC/TCP, degrees of freedom: 1), network (M1/M2/F1/F2, degrees of freedom: 3), limitation (unlimited/1 Mbps, degrees of freedom: 1). As the bandwidth limitation has a huge effect on the streaming performance (cf. Figures 1 and 2), additionally, separate two-factor ANOVAs were conducted for each bandwidth limitation condition. In these ANOVAS, only protocol and network are considered as independent variables.

The results of the ANOVAs for the different QoE factors are presented in Table III. It shows the F statistic and the p-value of the ANOVA. Three significance levels will be applied, and significant p-values will be highlighted and underlined according to their significance level in dark brown with a dotted line (5%), light brown/dashed (1%), or orange/solid (0.1%). Moreover, the effect size is quantified in terms of

TABLE III: Results of ANOVAs for all six QoE factors (three-factor, and two-factor for both bandwidth limitations).

	initial delay			number of quality changes			average resolution		
	F	p	ω_p^2	F	p	ω_p^2	F	p	ω_p^2
protocol	0.006	0.939	-0.002	0.017	0.895	-0.002	0.032	0.859	-0.002
network	0.686	0.561	-0.002	1.178	0.317	0.001	2.043	0.107	0.006
limitation	1083.688	$5 \cdot 10^{-126}$	0.682	373.088	$4 \cdot 10^{-62}$	0.425	1022.088	$9 \cdot 10^{-122}$	0.670
protocol*network	0.066	0.978	-0.006	0.963	0.410	$-2 \cdot 10^{-4}$	0.035	0.991	-0.006
protocol*limitation	0.012	0.913	-0.002	0.468	0.494	-0.001	0.400	0.528	-0.001
network*limitation	1.568	0.196	0.003	1.434	0.232	0.003	4.337	0.005	0.019
protocol*network*limitation	0.022	0.996	-0.006	0.701	0.552	-0.002	0.031	0.993	-0.006
bandwidth limitation: none									
protocol	0.706	0.407	-0.001	2.674	0.103	0.007	0.271	0.603	-0.003
network	15.554	$3 \cdot 10^{-9}$	0.149	0.559	0.643	-0.005	5.299	0.001	0.049
protocol*network	0.464	0.707	-0.006	0.559	0.643	-0.005	0.016	0.997	-0.012
bandwidth limitation: 1 Mbps									
protocol	$2 \cdot 10^{-6}$	0.998	-0.004	0.160	0.689	-0.003	0.113	0.737	-0.004
network	1.009	0.389	$1 \cdot 10^{-4}$	1.337	0.263	0.004	0.224	0.880	-0.009
protocol*network	0.041	0.989	-0.011	0.846	0.470	-0.002	0.057	0.982	-0.011

	average bitrate			number of stalling events			total stalling time		
	F	p	ω_p^2	F	p	ω_p^2	F	p	ω_p^2
protocol	0.001	0.977	-0.003	0.092	0.762	-0.002	0.609	0.435	-0.001
network	4.778	0.003	0.028	0.976	0.404	$-1 \cdot 10^{-4}$	0.340	0.796	-0.004
limitation	374.321	$2 \cdot 10^{-58}$	0.485	36.931	$2 \cdot 10^{-9}$	0.067	27.561	$2 \cdot 10^{-7}$	0.050
protocol*network	0.053	0.984	-0.007	0.268	0.849	-0.004	0.208	0.891	-0.005
protocol*limitation	0.067	0.795	-0.002	0.129	0.720	-0.002	0.701	0.403	-0.001
network*limitation	6.160	$4 \cdot 10^{-4}$	0.038	1.103	0.347	$6 \cdot 10^{-4}$	0.399	0.754	-0.004
protocol*network*limitation	0.032	0.992	-0.007	0.273	0.845	-0.004	0.168	0.918	-0.005
bandwidth limitation: none									
protocol	0.015	0.903	-0.005	-	-	-	-	-	-
network	5.334	0.002	0.064	-	-	-	-	-	-
protocol*network	0.033	0.992	-0.016	-	-	-	-	-	-
bandwidth limitation: 1 Mbps									
protocol	0.438	0.509	-0.003	0.092	0.762	-0.004	0.610	0.438	-0.002
network	0.461	0.710	-0.008	1.048	0.372	$6 \cdot 10^{-4}$	0.373	0.773	-0.007
protocol*network	0.478	0.698	-0.008	0.272	0.845	-0.009	0.190	0.903	-0.010

partial omega squared (ω_p^2), which maps the effect size to a range from -1 to 1. A small effect is indicated by values larger than 0.01 (dark brown with dotted line), a medium effect by values larger than 0.06 (light brown/dashed), and a large effect by values larger than 0.14 (orange/solid).

The results for initial delay are displayed in the top left part of Table III. The first seven rows show the results of the three-factor ANOVA, and thus, the effect of all three independent variables (protocol, network, limitation), all combined effect of two independent variables (protocol*network, protocol*limitation, network*limitation), as well as the combined effect of all three independent variables (protocol*network*limitation). It can be seen that only limitation shows a large effect. However, no effect of the transport protocol is visible ($p = 0.939, \omega_p^2 = -0.002$). The following lines display the results when the two limitation conditions are analyzed by separate two-factor ANOVAs. First, there are three lines with the results of the ANOVA for the unlimited streaming sessions for the two individual factors protocol and network, and the combined factor protocol*network. Here, only the network shows a significant, large effect on the initial delay. This contradicts the findings of the KS test, which showed a significant difference between QUIC and TCP for some networks. However, in the results of the ANOVA, the combined factor protocol*network is not significant. In case of the bandwidth limitation of 1 Mbps, the same factors are analyzed, and the results are shown in the last three rows. Here, all p-values are high, which means that no significant difference can be detected, which is in line with the KS test.

The ANOVA results for the number of quality changes (top middle), average video resolution (top right), and the average bitrate (bottom left) are next. For all three QoE factors, limitation has a large effect with very high values of ω_p^2. Network*limitation also showed small effects for average video resolution and average bitrate, just as network for average bitrate. When conducting the two-factor ANOVAs on the dataset for each limitation condition, only network showed a small effect on the average video resolution ($\omega_p^2 = 0.0049$) and a medium effect on the average bitrate ($\omega_p^2 = 0.064$) when the bandwidth was not limited. This effect is not visible for number of quality changes. Also no effect can be observed when the bandwidth was limited to 1 Mbps, which means that networks cannot be distinguished in terms of visual quality metrics if a bandwidth limitation is applied.

The bottom middle and bottom right part of the table show the results for the number of stalling events, and the total stalling time, respectively. In case of the three-factor ANOVAs, again, only the limitation is a significant influence factor, and shows a medium effect on the number of stalling events ($\omega_p^2 = 0.067$), and a small effect on the total stalling time ($\omega_p^2 = 0.050$). As stalling only occurred for a bandwidth limitation of 1 Mbps, only these results can be evaluated by a two-factor ANOVA. However, none of the investigated factors showed any effect on the stalling QoE metrics, like in the KS test.

Although protocol as an independent variable never showed a significant difference between the QUIC and TCP groups, a post-hoc analysis of the ANOVA results was conducted using Tukey's HSD test for the sake of completeness. It checks the null hypothesis that the means between the QUIC and TCP group are from the same population. As expected, for all QoE factors, including for the slightly controversial initial delay in the unlimited condition, the results of the test could not reject the hypothesis. This means that, in the conducted measurement study, there is no evidence that video streaming over QUIC and video streaming over TCP result in a different QoE.

V. Conclusion

This paper investigated the performance differences between video streaming over QUIC and video streaming over TCP, which are perceivable by end users and affect their QoE. An automated measurement framework was used to stream a defined set of videos from YouTube both via QUIC and TCP, while monitoring corresponding QoE factors (initial delay, number of stalling events, total stalling time, number of quality changes, average video resolution, and average bitrate). The measurements were conducted in four different access network using two bandwidth limitation conditions.

The impact of the three independent variables (protocol, network, limitation) on the six streaming QoE factors was analyzed statistically using a distribution-based and variance-based method. While bandwidth limitation obviously had significant effects on the QoE factors, the choice of network only exerted an influence when bandwidth was not throttled. This means that YouTube is able to deliver a homogeneous streaming experience over different networks when throughput is limited. However, regarding our main research question, i.e., the influence of the transport protocol (QUIC vs TCP), no evidence could be found that video streaming over QUIC and video streaming over TCP result in a different YouTube QoE. Given the performance gains promised, e.g., [13], this is a surprising result. It raises a number of questions regarding the general usefulness and applicability of QUIC for QoE management and optimization in today's and future networks, and poses new challenges how QoE management could be used to unlock the potential of QUIC to improve the QoE.

Nonetheless, further measurement studies are required to provide conclusive answers. Firstly, the range of tested application types has to be broadened, e.g., by including web and cloud applications as they are supposed to benefit from QUIC, too. Secondly, this study covered only network situations in which no or just a regular (1 Mbps) bandwidth limitation was applied. Thus, future measurements have to be extended to situations with more volatile network conditions. This, for example, includes configurations featuring additional packet loss, cross-traffic, or terminal mobility. It has to be investigated how the experience of Internet applications is affected by these more challenging network conditions and if QUIC is able to significantly improve QoE compared to TCP in these situations.

References

[1] J. Postel, "Transmission Control Protocol," Internet Requests for Comments, RFC 793, 1981. [Online]. Available: https://tools.ietf.org/html/rfc793

[2] M. Allman, V. Paxson, and E. Blanton, "TCP Congestion Control," Internet Requests for Comments, RFC 5681, 2009. [Online]. Available: https://tools.ietf.org/html/rfc5681

[3] C. Barakat, E. Altman, and W. Dabbous, "On TCP Performance in a Heterogeneous Network: a Survey," *IEEE Communications Magazine*, vol. 38, no. 1, 2000.

[4] G. A. Abed, M. Ismail, and K. Jumari, "A Survey on Performance of Congestion Control Mechanisms for Standard TCP Versions," *Australian Journal of Basic and Applied Sciences*, vol. 5, no. 12, 2011.

[5] J. Roskind, "QUIC: Design Document and Specification Rationale," 2012. [Online]. Available: https://docs.google.com/document/d/1RNHkx_VvKWyWg6Lr8SZ-saqsQx7rFV-ev2jRFUoVD34/edit

[6] S. Bouzas, "Why the Meteoric Rise of Google QUIC is Worrying Mobile Operators ," 2018. [Online]. Available: https://owmobility.com/blog/meteoric-rise-google-quic-worrying-mobile-operators/

[7] R. H. Alyssa Wilk and I. Swett, "A quic update on googles experimental transport," 2015. [Online]. Available: https://blog.chromium.org/2015/04/a-quic-update-on-googles-experimental.html

[8] J. Rüth, I. Poese, C. Dietzel, and O. Hohlfeld, "A First Look at QUIC in the Wild," in *PAM*, Berlin, Germany, 2018. [Online]. Available: https://quic.netray.io/stats.html

[9] M. Bishop, "Hypertext Transfer Protocol Version 3 (HTTP/3)," IETF WG QUIC, Internet-Draft, 2019. [Online]. Available: https://quicwg.org/base-drafts/draft-ietf-quic-http.html

[10] S. Egger, P. Reichl, T. Hoßfeld, and R. Schatz, "Time is Bandwidth? Narrowing the Gap between Subjective Time Perception and Quality of Experience," in *ICC*, Ottawa, Canada, 2012.

[11] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoßfeld, and P. Tran-Gia, "A Survey on Quality of Experience of HTTP Adaptive Streaming," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, 2015.

[12] J. Wagner, "Why Performance Matters," 2018. [Online]. Available: https://developers.google.com/web/fundamentals/performance/why-performance-matters/

[13] I. Swett, "QUIC – Deployment Experience @Google," Berlin, Germany, 2016. [Online]. Available: https://www.ietf.org/proceedings/96/slides/slides-96-quic-3.pdf

[14] D. Ghadiyaram, J. Pan, and A. C. Bovik, "A Time-varying Subjective Quality Model for Mobile Streaming Videos with Stalling Events," in *Applications of Digital Image Processing XXXVIII*, San Diego, CA, USA, 2015.

[15] M. Seufert, T. Hoßfeld, and C. Sieber, "Impact of Intermediate Layer on Quality of Experience of HTTP Adaptive Streaming," in *CNSM*, Barcelona, Spain, 2015.

[16] S. Egger, T. Hoßfeld, R. Schatz, and M. Fiedler, "Waiting Times in Quality of Experience for Web Based Services," in *QoMEX*, Yarra Valley, Australia, 2012.

[17] S. Cook, B. Mathieu, P. Truong, and I. Hamchaoui, "QUIC: Better for what and for whom?" in *ICC*, Paris, France, 2017.

[18] A. M. Kakhki, S. Jero, D. Choffnes, C. Nita-Rotaru, and A. Mislove, "Taking a Long Look at QUIC: an Approach for Rigorous Evaluation of Rapidly Evolving Transport Protocols," in *IMC*, London, UK, 2017.

[19] T. Zinner, S. Geissler, F. Helmschrott, and V. Burger, "Comparison of the Initial Delay for Video Playout Start for Different HTTP-based Transport Protocols," in *IM*, Lisbon, Portugal, 2017.

[20] D. Bhat, A. Rizk, and M. Zink, "Not so QUIC: A performance study of DASH over QUIC," in *NOSSDAV*, Taipei, Taiwan, 2017.

[21] V. Tong, H. A. Tran, S. Souihi, and A. Mellouk, "Empirical Study for Dynamic Adaptive Video Streaming Service Based on Google Transport QUIC Protocol," in *LCN*, Chicago, IL, US, 2018.

[22] A. Schwind, M. Seufert, Ö. Alay, P. Casas, P. Tran-Gia, and F. Wamser, "Concept and Implementation of Video QoE Measurements in a Mobile Broadband Testbed," in *MNM*, Dublin, Ireland, 2017.

[23] F. Wamser, M. Seufert, P. Casas, R. Irmer, P. Tran-Gia, and R. Schatz, "YoMoApp: a Tool for Analyzing QoE of YouTube HTTP Adaptive Streaming in Mobile Networks," in *EuCNC*, Paris, France, 2015.