2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX)

Is QUIC becoming the New TCP? On the Potential Impact of a New Protocol on Networked Multimedia QoE

Michael Seufert*, Raimund Schatz, Nikolas Wehner, Bruno Gardlo, Pedro Casas
AIT Austrian Institute of Technology GmbH, Vienna, Austria
**Now at: University of Würzburg, Würzburg, Germany*
michael.seufert@uni-wuerzburg.de, raimund.schatz@ait.ac.at, nikolas.wehner@ait.ac.at,
bruno.gardlo@ait.ac.at, pedro.casas@ait.ac.at

***Abstract*—Over the last years, QUIC (Quick UDP Internet Connections) has become the default protocol for networked communication of Google services, heralded as improved successor of the prevailing Transport Control Protocol (TCP). While the deployment of QUIC is increasing, QUIC is also planned to be the foundation of HTTP/3, the next generation of the HTTP protocols, which drive almost all applications on the Web. Given these developments, this paper aims to raise the awareness of the QoE research community to the increasing presence of QUIC, which likely brings implications for QoE monitoring and management of networked multimedia applications, as well as for the overall QoE research agenda. In particular, a major promise during the introduction of QUIC has been the improvement of the QoE of web-based applications (like browsing and video) by overcoming certain limitations and inefficiencies of TCP. In order to validate this claim, a measurement study was conducted to test whether the promised QoE benefits of QUIC are indeed noticeable for end users of streaming and browsing services. Surprisingly, no evidence for any QoE improvement of QUIC over TCP could be found. This way this paper aims to demonstrate how QoE research can and should successfully address relevant current and future developments on the Internet.**

I. Introduction

QUIC (Quick UDP Internet Connections) [1] is a new transport protocol, introduced by Google in 2013, which was designed to reduce the connection and transport latencies inevitable in the prevailing TCP (Transmission Control Protocol). It is based on UDP (User Datagram Protocol), but adds a cryptographic handshake to allow for zero round trip time (0-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 [2].

One advertised main reason behind the introduction of QUIC was the possibility to obtain a higher Quality of Experience (QoE) for customers of Google services, achieved by faster page load times [3] for web browsing based services like Google Search or Google Docs, and less stalling and a higher visual quality [4] 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 [5]. Google announced first results [6], [7] that QUIC makes page loading 5% faster on average and 1 s faster for web search at the 99 percentile. Moreover, according to [7], users reported 30% fewer stalling events for YouTube video streaming. Given these results, deliberate usage of the QUIC protocol classifies as effective form of QoE management.

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 [6], as it is the default protocol for communication between Google Chrome browsers and Google servers. 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. According to data gathered from 30 mobile operators [2], QUIC accounted for 20% of the total mobile traffic in November 2017, and its share was expected to further grow to 32% by November 2018. Recently, it was announced that HTTP over QUIC will become HTTP/3, the upcoming major version of the HTTP protocol [9]. Thus, QUIC adoption will expand beyond Google domains and affect an increasing number of web and cloud services. Consequently, research on QoE management and optimization has to consider it.

This paper investigates whether the usage of QUIC really yields noticeable QoE benefits to end users. Two measurement studies, one for video streaming and one for web browsing, have been conducted from the perspective of a naive end user, who just uses his or her private Internet access to stream videos or browse a web page. Perceivable QoE factors were monitored on application level, such as page load time, initial delay, the visual quality of the video, and stalling. Corresponding pairs of QUIC and TCP sessions were compared statistically in terms of QoE to find significant differences between streaming and browsing over QUIC versus over TCP.

978-1-5386-8212-8/19/$31.00 ©2019 IEEE

The remainder of the paper is structured as follows. Section II describes related work on QoE 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. Section V discusses implications for QoE research and concludes.

II. Background and Related Work

A. Video and Web QoE Metrics

Most works on QoE of (adaptive) video streaming agree that initial delay, stalling, and quality adaptation are the most dominant QoE factors [4]. Among these, stalling, i.e., the playback interruptions due to buffer depletion, is considered the worst QoE degradation [10]. Moreover, it is important to reach a high played out video quality [11], while the initial delay has only a small impact on the QoE [12]. Although different QoE models were proposed in literature, the recently standardized model P.1203 [13], [14] has gained a lot of attention and will also be used in the context of this work.

For Web QoE, response times were identified as the most important QoE factor [15]. This resulted in the development of first Web QoE models based on page load time (PLT), e.g., [12]. Recent research recognized the importance of the "above-the-fold" (ATF) time, i.e., the time until the visible portion of a web page has been fully loaded. This was also utilized to develop new Web QoE metrics, such as Google's Speed Index, which is the integral of the complementary visual progress that can be measured on the application level by inspecting the rendered pixel. However, such visual metrics might be inaccurate, when the looks of the page change, e.g., due to dynamic or asynchronously loaded contents. [16], [17] developed approximations for the SpeedIndex, which can be measured in the network. [18] combined traditional Web QoE models with ATF metrics and [19] employed machine learning for Web QoE models.

B. QUIC Performance Measurements

Apart from Google's report [7], which was mentioned above, few works have considered the impact of QUIC on QoE. [20] measured PLTs 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. [21] extended previous works, which mainly focused on PLT experiments. They found that QUIC could outperform TCP in terms of PLT for most scenarios due to the 0-RTT feature. They also investigated video streaming performance in controlled environments over high-speed links with small packet loss and found that QUIC can outperform TCP for video streaming only for high resolutions. [22] found that QUIC achieved shorter initial delays for YouTube than TCP especially with increasing RTT or packet loss, but only when leveraging 0-RTT connection setup. [23] found that QUIC provided better QoE for HAS over uncontrolled wireless network environments in the public Internet. In [24], the impact of TCP vs QUIC on QoE factors of video streaming was investigated in a home setup from a

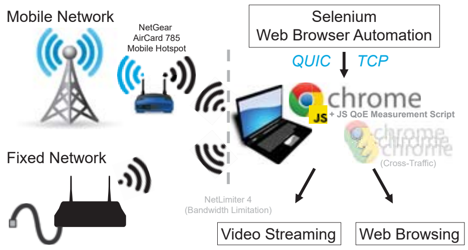


Fig. 1: Measurement setup

naive end user QoE perspective, i.e., no control over network, browser, or server settings was applied. From the evaluation of the measurement study, no significant differences between the groups of QUIC and TCP sessions were found for any QoE factor. In this work, the study in [24] will be extended with more detailed evaluations for video streaming based on paired comparisons of QUIC and TCP sessions. Moreover, this work additionally investigates the impact of QUIC vs TCP for web browsing sessions, which was not covered in [24].

III. Methodology

The goal of the study is to investigate the impact of the transport protocol (QUIC vs TCP) on the QoE for web browsing and video streaming from the perspective of a naive end user. The measurement framework of [24] was extended, which used the Selenium browser automation library to automatically start a Chrome browser and browse to specific webpages. JavaScript-based monitoring scripts were injected into the page to obtain application-level QoE factors. Moreover, the chrome browser was configured such that all HTTP requests were logged to a file (`-log-net-log`), and all network traffic was captured using tshark.

Each webpage was visited twice, once with QUIC traffic enabled in the Chrome browser (`--enable-quic`), and once with QUIC traffic disabled, i.e., the data were transmitted using TCP. Both 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 sessions should face similar network conditions, although this influence factor was not controlled in the measurement setup. For each pair of sessions, the order of the 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 measurement framework was installed on a laptop in a home setup. The measurements were taken in four different customer access networks to obtain more generalizable results, i.e., in a mobile network in Country 1 (M1), roaming in a mobile network in Country 2 using the same Country 1 SIM card (M2), and two fixed home network in Country 1 (F1) and Country 2 (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 Country 1 SIM card. For the measurements in the fixed networks, the measurement laptop was directly connected to the home router via Wi-Fi. Table I

TABLE I: Maximum link speeds in considered mobile/fixed access networks measured by online speed tests.

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

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 pairs of sessions, a bandwidth limitation to 1 Mbps on both down- and uplink was applied using the tool NetLimiter 4 on the measurement laptop. The other pairs were streamed without any bandwidth limitation (unlimited). Note that QUIC and TCP sessions belonging to the same pair were streamed using the same conditions in terms of network and bandwidth limitation.

A. Video Streaming

Over a period of several days in October/November 2018, pairs of YouTube video sessions were streamed and recorded in all four networks. For that, random YouTube video IDs were generated and each was streamed with both QUIC and TCP without any additional traffic on the access link. Every 250 ms, the monitoring script [25] recorded the current timestamp, the current video playtime, buffered playtime, video resolution, and player state, and logged all QoE factors to a file. Inspecting the video flows made sure that the videos were either streamed via TCP or QUIC, just as configured by the measurement application. Moreover, the logged streaming information were inspected to ensure that both the corresponding QUIC and TCP sessions did not contain any advertisement clip.

The final data set consisted of 252 pairs of YouTube sessions 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. The left part of 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 pairs of QUIC and TCP sessions, in which the same video was streamed. 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. Moreover, a QoE score was computed using the standardized P.1203 QoE model [13], [14].

TABLE II: Number of measured streaming and browsing sessions by access network and bandwidth limitation.

Video Streaming			Web Browsing		
network	unlimited	1 Mbps	network	unlimited	1 Mbps
M1	58	52	M1	1108	1134
M2	42	62	M1 (CT)	978	852
F1	62	50	F1	1112	1034
F2	88	90	F1 (CT)	986	998

B. Web Browsing

In addition to video streaming, also web browsing was investigated. In December 2018 and January 2019, pairs of browsing sessions were measured and recorded with the framework in a mobile network (M1) and a fixed network (F1). The top 67 most popular webpages[1] were selected. However, as only the websites of Google and YouTube could be browsed with both QUIC and TCP – the other pages only supported TCP – only these services could be analyzed. The browsed pages include the google.com and youtube.com main page, Google landing pages for other top-level domains, as well as Google and YouTube search result pages for search queries, which were accessed using a direct URL with one of 50 random search terms. In total, 4101 pairs of browsing sessions were measured. For roughly half of the sessions, cross-traffic (CT) was added on the access link, to investigate whether QUIC or TCP had advantages when directly competing with other flows on the link. This means, three other QUIC or TCP browsing sessions were started on the measurement laptop 100 ms before, at the same time, and 100 ms after browsing to the measured webpage. As the presence of CT on the access link will significantly reduce and vary the available bandwidth for the measured browsing session, it will be considered as a different network, i.e., M1 (CT) and F1 (CT).

This results again in a factorial design with three independent variables, i.e., protocol, network, and limitation, as shown in the right part of Table II. Note again that for each pair of QUIC and TCP sessions, the browsed webpage and the other independent variables (network, limitation, cross-traffic) were identical. The injected monitoring script queried the Navigation Timing API and logged all performance entries. The time until the DOMComplete event was fired by the browser was selected as the dependent variable, i.e., the page load time (PLT) of the browsing session. Although other metrics were suggested, such as time to interactive (DOMInteractive event, document loaded without CSS or images) or time until the ATF part of the page is fully loaded (might be inaccurate for dynamic or asynchronously loaded contents), the DOMComplete event was used here. It marks the time when the page and all sub-resources are fully loaded, which is indicated by the disappearance of the spinning loading icon, and thus, quite noticeable for end users. Finally, a QoE score was computed from the PLT using the WQL PLT model proposed in [18], i.e., $QoE \sim \log(PLT)$.

[1] https://en.wikipedia.org/wiki/List_of_most_popular_websites, source: top 50 lists published by Alexa Internet and SimilarWeb, accessed: 12/12/2018.

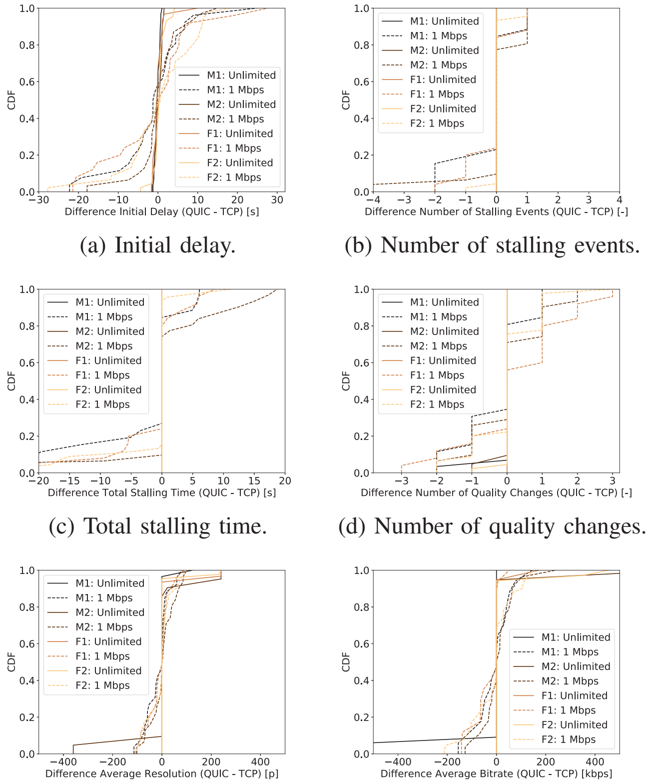
IV. Evaluation of Impact on QoE Factors

This section investigates the impact of the protocol on the QoE factors. In particular, the differences within the pairs of corresponding QUIC and TCP sessions are analyzed. Note again that in this work, for both video streaming and web browsing, Google services on Google servers are accessed, just like an end user would. Assuming that Google has already optimized the service delivery, this avoided the need for calibration of own QUIC servers, e.g., [21]. Also, using real customer Internet access, no artificial network artifacts like packet loss or delay had to be added, which was necessary in other more controlled performance studies, e.g., [20]. Thus, the presented results below shed light on QUIC vs TCP *in the wild*, i.e., they present the situation *as is* for naive end users.

A. Video Streaming

For the investigated video streaming sessions, cumulative distribution functions (CDFs) of the QoE factors were already shown in [24] and it was reported that both Kolmogorov-Smirnov (KS) tests and analyses of variance (ANOVAs) could not find any evidence that video streaming over QUIC and video streaming over TCP result in different QoE factors. The evaluations only confirmed that the bandwidth limitation had the strongest effect on the QoE factors. As the previous evaluations considered differences between the whole groups, these evaluations will be continued by focusing on pair-wise statistical difference analysis. This means that this works extends [24] by considering a more strict paired comparison of QUIC vs TCP for video streaming. For each of the QoE factors, the difference between the value of the QUIC session and the value of the corresponding TCP session is considered. Figure 2 shows the CDFs of these differences for initial delay, number of stalling events, total stalling time, number of quality changes, average video resolution (in vertical pixels [p]), and average bitrate. Each plot in Figure 2 shows eight CDFs of differences. Here, the different networks are distinguished by color from orange (M1), light brown (M2), dark brown (F1) to black (F2), and the two limitation conditions unlimited (solid) and 1 Mbps (dashed) are distinguished by line style.

For all six QoE factors, the CDFs of differences are symmetric around 0. The CDFs in Figure 2a indicate an almost uniform distribution of the difference of initial delays, i.e., the difference between the initial delay of the QUIC session and the initial delay of the TCP session. Here, the differences are spread over a larger interval for the 1 Mbps limitation condition. For the remaining QoE factors, also a little spread can be observed for the 1 Mbps limitation, while for unlimited bandwidth the differences are mostly 0 for most of the sessions. The differences between each pair of sessions were also investigated statistically by conducting a paired t-test to check for significant differences on a significance level of 5%. For all QoE factors, the null hypothesis that the mean of the difference is 0 could not be rejected, even when distinguishing between both limitation conditions and all four networks. This means, no significant difference of application-level QoE factors between QUIC and TCP could be observed.



(a) Initial delay.

(b) Number of stalling events.

(c) Total stalling time.

(d) Number of quality changes.

(e) Average resolution.

(f) Average bitrate.

Fig. 2: Distributions of differences between QoE factors measured in QUIC sessions and their corresponding TCP sessions for different access networks and bandwidth limitations.

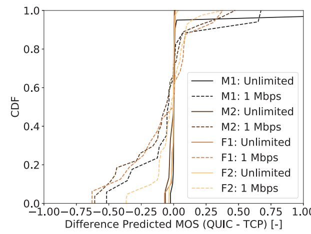
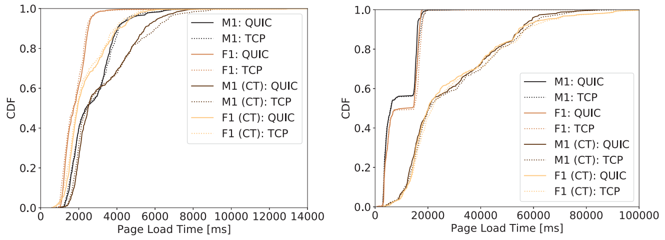


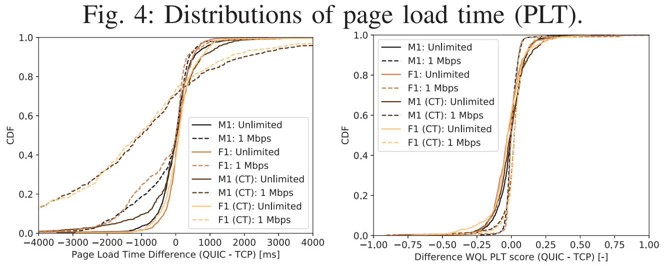
Fig. 3: Distributions of differences between P.1203 scores calculated for corresponding QUIC and TCP video sessions.

For all sessions, a QoE score was computed using the standardized P.1203 model [13], [14], which indicates the QoE on a continuous scale from 1 (bad) to 5 (excellent). Figure 3 depicts the CDFs for the differences of the P.1203 score within the pairs of video streaming sessions. Again, for unlimited bandwidth, the CDFs are almost vertical, but for a limitation of 1 Mbps a small symmetric spread can be observed. In the corresponding t-test over all pairs of sessions, there is no significant difference on the P.1203 score between QUIC and TCP. When distinguishing the limitation conditions, only the unlimited bandwidth condition showed a significant difference for the P.1203 score. However, the mean difference here is -0.04 on the P.1203 scale, which can be neglected in terms of QoE. Thus, no impact of the transport protocol can be found when comparing the QoE of the same video sessions.



(a) Unlimited. (b) 1 Mbps.

Fig. 4: Distributions of page load time (PLT).



(a) Measured PLT. (b) Calculated WQL PLT scores.

Fig. 5: Distributions of differences between corresponding QUIC and TCP web browsing sessions.

B. Web Browsing

Figure 4 shows the CDFs for the page load time (PLT) in all web browsing sessions for both unlimited bandwidth and a bandwidth limitation of 1 Mbps. The different networks are visualized by different colors, namely, M1 (black), F1 (light brown), M1 (CT) (dark brown), and F1 (CT) (yellow). The PLTs of QUIC sessions are plotted with solid lines and the corresponding TCP sessions with dotted lines. It can be seen that the PLTs are well below 10 s for unlimited bandwidth, but they can extend up to 100 s in case of a bandwidth limitation to 1 Mbps plus additional cross-traffic. For some conditions, e.g., for M1 or F1 for 1 Mbps, it is nicely visible that the PLTs follow a bimodal distribution, which is caused by the different complexity of Google and YouTube search result pages. Moreover, it can be seen that the CDFs of QUIC sessions and their corresponding CDFs of TCP sessions are very close, which indicates that the distributions are similar. Only for some conditions, namely, M1 (CT) with unlimited bandwidth, and M1 and F1 for 1 Mbps, the KS-test rejects the null hypothesis that the distributions are identical for a significance level of 5%. Nevertheless, ANOVA found no significant differences between the groups of QUIC and TCP sessions. Thus, the differences within a pair of sessions have to be investigated in more detail in terms of PLT and QoE.

Figure 5a presents the CDFs for the differences within a pair of sessions, i.e., the PLT of the QUIC session minus the PLT of the corresponding TCP session. In the unlimited conditions, the CDFs again look symmetric around 0 with only small deviations. However, for 1 Mbps, it can be seen that the distributions are shifted more towards negative differences, which indicate the QUIC sessions have a shorter PLT. Especially, for 1 Mbps conditions with cross-traffic, it is clearly visible that around 75% of the pairs of sessions have a negative difference. When conducting the t-test, a significant difference can be found. Here, the mean difference in PLT is -365 ms for all sessions. When separating the different limitation conditions, no significant difference can be found for unlimited bandwidth. However, for 1 Mbps the difference is significant and has a mean value of -762 ms. This suggest advantages of QUIC in low bandwidth conditions, however, as observed in Figure 4b, absolute PLTs are already very high here. Thus, the QoE might nevertheless be low.

To quantify the impact of the PLT in terms of QoE, the WQL PLT model from [18] is used, namely, $QoE = -0.5368 \log(PLT) + 7.9035$. It maps the PLT to a QoE score on a continuous scale, where 1 indicates bad and 5 indicates excellent QoE. Figure 5b presents the CDFs of the differences of the WQL PLT score within a pair of sessions. Again, very symmetric CDFs can be found, which are located around 0 with little spread. When conducting the t-test for all sessions, also no significant differences can be found. Only for both limitation conditions individually, the tests indicate significant differences. However, in both cases, there are only marginal mean differences of -0.01 (unlimited) and 0.01 (1 Mbps) for the WQL PLT score, which can be neglected in terms of the QoE perceived by end users. Thus, also no impact of the transport protocol can be found when comparing the QoE of the same browsing sessions.

V. Discussion and Conclusion

The previous section presented the results of our measurement studies to compare the performance of QUIC and TCP on application- and QoE-level. As regards video streaming, we could not find any significant performance gain of QUIC over TCP (despite having measured a high number of sessions), except for conditions without bandwidth constraints where statistically significant QoE improvements were detected. However, even in these cases, the mean calculated QoE score difference (0.04 based on P.1203) actually is too low to be perceivable at all. Thus, one can safely conclude that no QoE-relevant improvement by QUIC could be found for video streaming. Also in the case of web browsing, where QUIC exhibited PLT performance gains only with bandwidth limitations, the mean QoE score differences are too small (0.01) to be relevant. We conclude from these results that we could not detect any QoE-relevant advantages of QUIC over TCP, contradicting the claims in related work like [6], [7]. In different words: according to our results, the use of QUIC instead of TCP *per se* does not yield any noteworthy QoE improvements.

These results raise a number of questions regarding the validity and generalizability of the results. In this respect, our measurement study was limited in terms of volume and scope due to practical constraints. One could argue that only YouTube videos were streamed and the YouTube and Google (search) pages were browsed for testing. However, these services represent a considerable share of Internet traffic, they were claimed to significantly benefit from QUIC (cf. [6], [7]), and we can assume that Google's implementation of QUIC is technically correct and sufficiently optimized. Nonetheless, we see a necessity for future studies that benchmark TCP vs QUIC using a wider variety of content (e.g., featuring a wider variety of technical properties like number and size of web objects) and services (e.g., Google Docs or Google Maps) in order to obtain a more comprehensive overview of configurations where QUIC might truly outperforms TCP in terms of QoE. Furthermore, one could object that our measurements were conducted *in the wild*, i.e., using Google services accessed via the public Internet and different ISPs' access networks. While such an approach generally tends to increase levels of noise and variability of measurements compared to a controlled lab testbed (e.g., [21]), it is the more ecologically valid option since our goal was to assess QoE improvements for real users under real-world conditions. In addition, we addressed this aspect by using a fairly high number of measurements, which can be expected to be high enough to provide our study with sufficient statistical power. Still, we see value in conducting similar measurement studies that feature a controlled setup with self-hosted web applications in order obtain a further reference and more conclusive answers on the technical reasons behind the presented results, which could be similar congestion control algorithms used by TCP and QUIC, ineffective implementations of 0-RTT, or suboptimal configurations of QUIC.

Finally, there remain open issues regarding the general usefulness and applicability of QUIC for QoE management and optimization in today's and future networks. In general, our study demonstrates the importance of including an end-user QoE perspective in evaluating new technologies and changes proposed to the Internet ecosystem. We could show that – beyond latency and throughput – not only choosing the right user-facing application-level metrics (like PLT, initial delay, stalling) can provide a solid basis for benchmarking, but also that the application of QoE models (like WQL, P.1203) leads to further relevant insights, as they cover interactions between QoE factors and their mapping to human perception. Still, the fact that we could not find any evidence that switching from TCP to QUIC (in its current implementation) in practice leads to noteworthy QoE improvements, leaves us with the question of how QoE and networking research could be used to unlock the potential of QUIC to truly improve the QoE of web-based services. On the one hand, QoE testbeds (like [21]) and field measurement setups (like the one presented in this paper) could be used not only for benchmarking, but also for promoting QoE-centric tuning of the different performance and implementation parameters of the QUIC protocol (balanced with other relevant aspects like bandwidth efficiency and robustness). On the other hand, the fact that QUIC is an open source application-layer protocol could be leveraged for implementing and trialling advanced QoE management approaches (like SAND [26]) that can be utilized in existing operating and experimental future networks. The goal would be to consistently achieve – also in the wild – the promising QoE improvements by QUIC, which some other studies have found. This way, the multimedia quality research community could take on an even more visible and active role in realizing the vision of a QoE-enabled Internet.

References

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

[2] 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/

[3] 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*, 2012.

[4] 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.

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

[6] 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

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

[8] J. Rüth, I. Poese, C. Dietzel, and O. Hohlfeld, "A First Look at QUIC in the Wild," in *PAM*, 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] 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*, 2015.

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

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

[13] ITU-T, "Recommendation P.1203: Parametric Bitstream-based Quality Assessment of Progressive Download and Adaptive Audiovisual Streaming Services over Reliable Transport," 2016.

[14] M. Seufert, N. Wehner, and P. Casas, "Studying the Impact of HAS QoE Factors on the Standardized QoE Model P. 1203," in *Internet-QoE*, 2018.

[15] J. Nielsen, *Usability Engineering*, 1993.

[16] E. Bocchi, L. De Cicco, and D. Rossi, "Measuring the Quality of Experience of Web Users," *ACM SIGCOMM Computer Communication Review*, vol. 46, no. 4, 2016.

[17] M. Trevisan, I. Drago, and M. Mellia, "PAIN: A Passive Web performance indicator for ISPs," *Computer Networks*, vol. 149, 2019.

[18] T. Hoßfeld, F. Metzger, and D. Rossi, "Speed Index: Relating the Industrial Standard for User Perceived Web Performance to Web QoE," in *QoMEX*, 2018.

[19] Q. Gao, P. Dey, and P. Ahammad, "Perceived Performance of Top Retail Webpages in the Wild: Insights from Large-scale Crowdsourcing of Above-the-fold QoE," in *Internet-QoE*, 2017.

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

[21] 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*, 2017.

[22] 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*, 2017.

[23] S. Arisu and A. C. Begen, "Quickly Starting Media Streams Using QUIC," in *PV*, 2018.

[24] M. Seufert, R. Schatz, N. Wehner, and P. Casas, "QUICker or not? - an Empirical Analysis of QUIC vs TCP for Video Streaming QoE Provisioning," in *QoE-Management*, 2019.

[25] 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*, 2015.

[26] ISO/IEC, "23009-5:2015 Information Technology – Dynamic Adaptive Streaming over HTTP (DASH) – Part 5: Server and network assisted DASH (SAND)," 2016.