The simulation study on the multipath adaptive video transmission

Slawomir Przylucki[1],*, and *Dariusz* Czerwinski[1]

[1]Lublin University of Technology, Faculty of Electrical Engineering and Computer Science, Institute of Computer Science, Nadbystrzycka 36B, 20-618 Lublin, Poland

Abstract. The article presents simulation studies of multipath video transmission implemented as a live monitoring system based on adaptive streaming mechanisms. The transmission network consists of the Wi-Fi infrastructure and a single LTE cell. Both wireless networks are used during the operation of the video monitoring system. The analysis of the obtained results allows us to compare the features of two currently used multipath transmission protocols, MultiPath Quick UDP Internet Connections (MPQUIC) and MultiPath TCP (MPTCP) in the context of their use in adaptive video streaming systems. Also, the article contains a summary of the perceived advantages and disadvantages of using several transmission networks simultaneously for the implementation of video surveillance and monitoring systems.

1 Introduction

Nowadays, most video transmission systems in industry are based on shared IP networks. This solution offers flexibility and low implementation and maintenance costs. On the other hand, the quality of services, such as video monitoring of technological processes or broadly understood video surveillance, is closely related to the possibility of providing sufficient bandwidth for a video stream with specified parameters. The last few years have opened the possibility of solving the quality issue owing to the development of the HTTP Adaptive Streaming (HAS). One of the most popular implementations of this standard is Dynamic Adaptive Streaming over HTTP (DASH) [1]. The basic principle of DASH is the replacement of progressive download of a video by the more sophisticated approach, based on a qualitatively differentiated, segmented video representation. From the implementation point of view, each video content is segmented into chunks. A single chunk represents a video segment of specified quality. By creating many chunks with different quality, DASH allows the client to choose the best representation according to his technical parameters but also the network conditions [4]. The selection of the representation (chunk) is determined by the rate adaptation algorithm implemented on the client side [3].

Unfortunately, the need to prepare such a specific, multi-representations of video before distributing them to clients, definitely limited the use of DASH in live transmission systems. The situation changed with the publication of the extension of the DASH in the form of Server and Network-Assisted DASH (SAND) standard [2]. The purpose of SAND is to provide a new toolset supporting video content delivery. At the same time, the system addressed the issue of accuracy of methods of adaptation and reaction time of the video streaming system. SAND's foundation is the ability to communicate between elements of the system to better predict network fluctuations and guarantee adequate video content quality on the client side. The standard defines a new class of devices, DASH-Aware Network Elements (DANE) and Metric server. It also introduces the following messages for exchanging information between components of the video streaming system [2]:

- between DANEs - Parameters Enhancing Delivery (PED),
- from DANEs to clients - Parameters Enhancing Reception (PER),
- from clients to DANEs - Status messages,
- from clients to Metrics server: Metrics messages.

These messages notify the video server (also in advance) about the required chunks, hence there is no longer any need for creating redundant representations. Therefore, the system based on SAND can work in the same way as conventional monitoring systems or in general, conventional live transmission system. However, it still retains the adaptation feature offered by HAS systems [5].

At this point, it should be emphasised that despite the advantages offered by SAND systems, it is still crucial to ensure sufficient bandwidth for the implementation of video services. This will minimise the number of decisions to downgrade the quality of downloaded video representation. In this context, the multipath transmission appears to be a promising method to complement the advantages of adaptive streaming. The technology in question is currently attracting great attention due to the increasing availability of multihomed devices, *i.e.* devices that can simultaneously use various data transmission techniques. Naturally, the dedicated transmission protocol is a necessary complement to any multipath system. The most promising representatives of the protocols of this kind are: MultiPath Quick UDP Internet Connections

[*] Corresponding author: s.przylucki@pollub.pl

(MPQUIC) and MultiPath TCP (MPTCP). We think it is worth to compare both protocols experimentally and assess how useful they are in streaming services based on the standard DASH. In other words, the premise for our research is to indicate the advantages and disadvantages of combining both techniques, *i.e.* DASH standard and multi-path transmission in the typical video streaming system.

The remainder of the paper is organised as follows. The multipath TCP extensions and works related to their use in video streaming systems are presented in Section II. Section III contains the description of the proposed test scenario. Section IV includes the presentation of results obtained and their comparative analysis. Finally, a brief conclusion is made in Section V.

2 Background and related works

Over the last few years, a great amount of research results have appeared that assessed new proposals on improving the transport layer. Among them, two new standards are directly related to the issue of simultaneous data transmission across multiple paths:

- Google QUIC protocol – extension of UDP transport, which implements TCP-like features at the application layer [6],
- Multipath TCP (MPTCP) – the TCP extension that allows using multiple paths in data transmission [6] [7].

Both modifications of transport protocols offer potential improvement in the quality of received video streams. However, the usage of them affects differently the performance of DASH applications due to their adaptive nature.

2.1 Multipath TCP extension

MPTCP is a solution built on top of the TCP protocol which defines data transport over multiple TCP sessions [7]. The implementation of MPTCP assumes that no additional mechanisms at the application level are required. In other words, end-user applications are unaware that the transmission is based on two or more sub-flows. The idea of sub-flows is crucial in this context because MPTCP creates sub-flows for each interface and offers dedicated scheduling mechanisms which distribute the data segments among them [9]. In this way, MPTCP enables increasing data transfer efficiency using multiple interfaces.

Quick UDP Internet Connections technology (QUIC) represents completely different approach than MPTCP. It is implemented at the application layer and it is strictly coupled with UDP transport layer network protocol [11]. The present implementation of QIUC offers encrypted end-to-end connections and thus protection of not only the application-layer content but also the transport-layer headers. The main premise for the creation of QUIC was the willingness to provide the reduced connection and transport latency along with security and reliability [10]. Nowadays, the source code of QUIC is public and the project is strongly supported by Google.

Until 2017, no multipath extension of protocol QUIC had been proposed. Even the most up-to-date versions of this protocol, such as SCTP, support only several streams. In [12], authors proposed a multipath solution named Multipath QUIC (MPQUIC). It is based on the same premises that formed the basis for the creation of the MPTCP and it offers an effective solution to the problem of path identification and managing multiple paths [13]. The implementation of MPQUIC was also made available to the public in the form of source code along with a set of useful tools. This allows researchers not only to evaluate the protocol but also to compare its properties, *e.g.* with MPTC [19].

2.2 Multipath approach in systems DASH

Both solutions described above belong to a relatively new group of transmission protocols. For this reason, there is a very limited number of publications which relate to the implementation of DASH video streaming in the multipath-aware environment.

Among the most-cited works, it is worth mentioning the studies devoted to the MPTCP performance over wireless networks [14], [15]. They clearly prove that MPTCP offers the mechanism of sending packets (also with video content), which significantly increasing streaming application performance. However, on the other hand, it was observed that the variability of available bandwidths among MPTCP sub-flows affected the perceived quality offered by adaptive streaming systems such as DASH. Also, the rebuffering and suboptimal decisions regarding the section of video quality have been reported [17]. In [16], authors considered the wireless video transmission and analysed the impact of variability of the available bandwidth on the quality of MPTCP-based video services. The conclusions from this work confirmed the observations contained in [14].

The second of the multipath solutions, MPQUIC, was presented in 2017 and so far, (to the best of our knowledge at the time of this publication) it has not been analysed in the context of DASH services.

3 The testing procedure

The basic assumption of the performed experiments was to implement both multipath transmission protocols in a way which would ensure the possibility of a direct comparison of DASH system behaviour. We used the most up-to-date version of the MPTCP protocol [18] and on the only currently available version of MPQUIC (apart from the tools offered on Google servers) [19]. The configurations of both protocols are based on the default settings and parameter values, according to the recommendations contained in [8] [20] [19].

The network topology of the testbed environment was based on virtualisation (KVM hypervisor). The basic hardware platform for the test environment was the server (CPU Intel(R) Core (TM) i7-6700 3.40GHz, 32GB RAM, 1TB SSD RAID10). Well-known Linux kernel tools, netem and tc were used to emulate network conditions including available bandwidth, loss, delay, *etc*.

3.1 The testbed configuration

The DASH server (video server) was implemented on the basis of Apache server (2.4.29). It was installed on two virtual machines (Ubuntu 16.04) with identical parameters (4 CPU cores, 8 GB RAM, interfaces driver: virtio). In case of MPTCP environment (VM3), the kernel was compiled with MPTCP extensions. For QUIC protocol tests (VM4), the dedicated QUIC server was used. It served as proxy (QUIC connection - HTTP connection). In addition, the kernel was supplemented with an extension MPQUIC go.

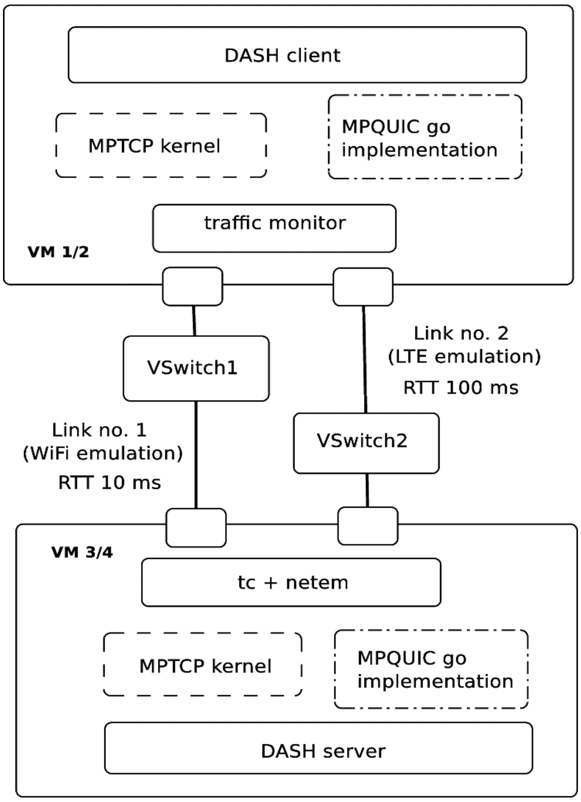

Fig. 1. The structure of testbed.

The DASH clients (also two VMs, VM1/2) based on Linux with Google Chrome browser and Shaka-player [21]. Both machines had identical parameters (2 CPU cores, 4 GB RAM, interfaces driver: virtio). The kernels of the operating systems of both virtual machines were also prepared analogically to servers. Two transmission paths were defined between DASH server and client (for each multi-path transmission environment under investigation). They were configured in a virtual environment, based on VSwitch solution and emulated multihomed behaviour. The structure of the developed testbed is presented in Figure 1.

3.2 The scenarios of the tests

Two transmission paths are used to emulate a typical configuration of multihomed devices. We have assumed in all tests that the main connection is Link no. 1, and Link no. 2 is a secondary link. The Round-Trip Times (RTT) for Link no.1 was 10 ms and for Link no.2 was 100 ms. These values correspond to the typical values observed in real networks, *e.g.* Wi-Fi 802.11n (Link no.1) and single cell LTE-A (Link no.2).

To simplify the test environment, it was assumed that the server has all the necessary video representations (no real-time coding is needed). This simplification, according to the information contained in the previous part of the article, does not affect the assessment of the functioning of the DASH / SAND system. For this reason, the video content provided on the DASH server came from a publicly available database published by ITEC [22]. We chose the video titled Elephant Dream. This video has 20 representations that differ in coding parameters and thus the required bandwidth. The top 6 representations (from the point of view of quality) required bandwidths of 2.1, 2.4, 2.9, 3.3, 3.6 and 4.0 Mbps respectively. Each video representation was divided into 40 chunks with 15-second duration.

Our experiment consists of three tests. Their purpose was to analyse the behaviour of the tested protocols at various bandwidth variability scenarios on two links forming the video transmission path. As a reference value for the bandwidth available on an individual link, we assumed the bandwidth required by the best video representation (4 Mbps). Throughout the paper, this value is defined as Bmax. The scenarios of the tests are presented in Table 1.

Table 1. A set of parameters used in tests.

Scenario	**Bandwidth**	
	Link no. 1	Link no. 2
Test no.1	const. 0.6 Bmax	var. 0.0 – 0.4 Bmax
Test no. 2	var. 0.4 – 0.8 Bmax	const. 0.2 Bmax
Test no. 3	var. 0.4 – 0.8 Bmax	var. 0.0 – 0.4 Bmax

4 The analysis of the results

During test no.1, the available bandwidth of the Link no.1 was constant but smaller than required by the best representation of the streaming video (Bmax). The use of the second link should improve the streaming conditions by allowing the download of chunks at a higher rate. According to Figure 2A, two episodes of reducing the available bandwidth (60 s – 105 s and 480 s – 525 s) appeared during the test and single time interval in which the total available bandwidth exceeded the bandwidth required by the streaming content (230 s - 285 s). Between them, there were periods of relatively stable transmission conditions but with different levels of total bandwidth (120 s – 225 s and 330 s – 405 s).

Comparing both algorithms (Fig. 2B and 2C), it can be noticed that MPQUIC tried to use the bandwidth available on Link no.2 more aggressively. This phenomenon also occurred during periods of stable conditions of transmission. At the same time, MPQUIC algorithm very well followed the changes in the available bandwidth (in the overload periods as well as in the range of the largest, available total throughput). These conclusions are also confirmed by data presented in Table 2 and Table 3.

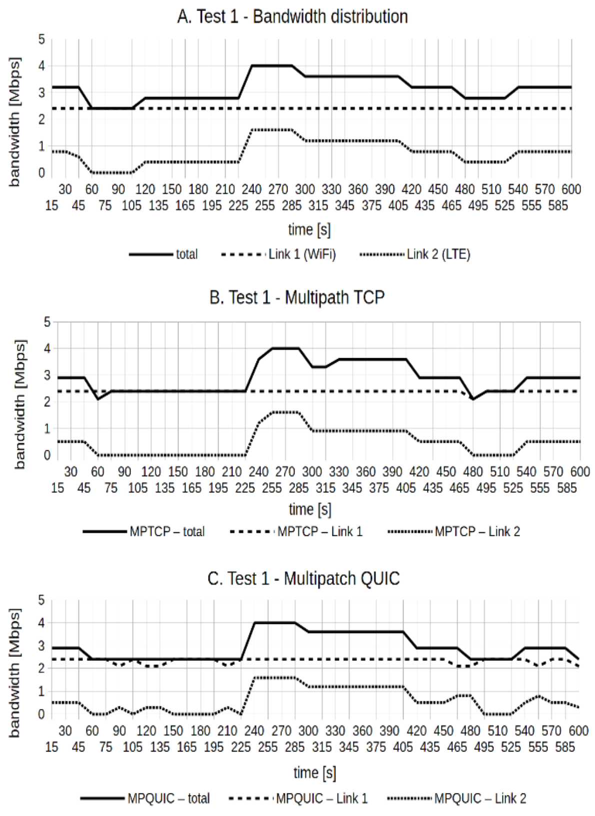

Fig. 2. Bandwidth distribution during test no. 1.

In Table 3, the "avg." means the average values of the usage of available bandwidth in 100 independent measurements and "dev" column consists of the average values of the absolute deviations of data points from their mean. Based on this data, it can be stated that the MPTCP protocol allowed obtaining a slightly higher value of average bandwidth usage during the test. In addition, this was achieved with smaller fluctuations, which positively affects the functioning of caching algorithms implemented on the client's side.

Table 2. Test no.1 – number of switches (average values).

Protocol	Number of switches		
	All	Down	Up
MPTCP	9	4	5
MPQUIC	7	5	2

This is also confirmed by the number and type of switches within the DASH adaptation algorithm. The MPTCP generated fewer switching events and fewer downgrades (switching down**).** In order to objectively assess this impact of switching, the values of the quality indicator in the form of PSNR (Pseudo Signal to Noise Ratios) were compared. The PSNR values were calculated from the algorithm implemented in the psnr tool of libyuv. The values represented quality measure per frame and then these values ware averaged for each fragment (chunk) of the video being streamed. The discrepancy between the quality of the original video and received at the time of the highest degradation (switching to the lowest bit rate) was 12.6%.

Table 3. Test no.1 – usage of available bandwidth.

Protocol	Bandwidth used					
	total		Link no. 1		Link no.2	
	avg.	dev	avg.	dev	avg.	dev
MPTCP	2.96	0.433	2.39	0.014	0.51	0.375
MPQUIC	2.94	0.466	2.34	0.094	0.61	0.453

In the next test, the fixed value of the available bandwidth was assigned to Link no.2 (Figure 3). Again, the use of the MPQUIC protocol was associated with larger fluctuations in the bandwidth used on both links. This is clearly seen when comparing Figure 3B and 3C, as well as the data in Table 4 and 5.

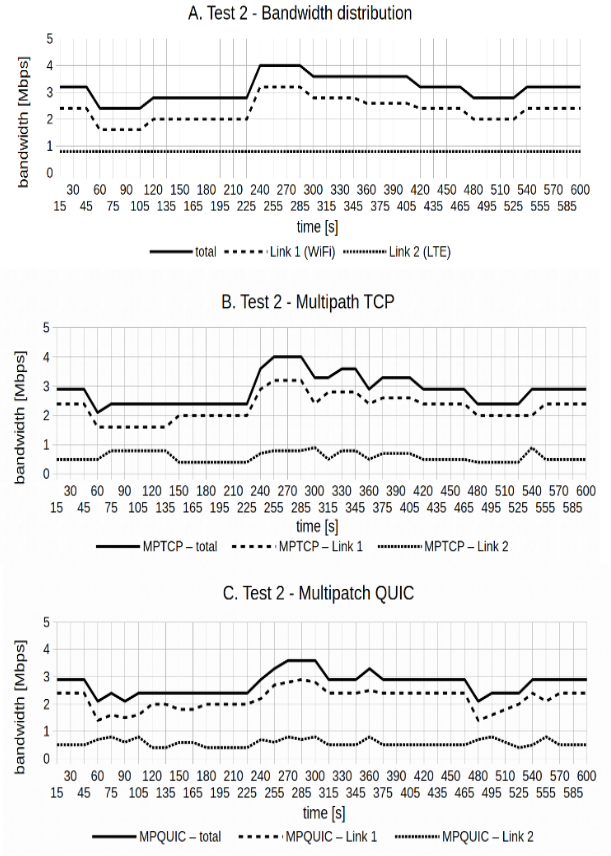

Fig. 3. Bandwidth distribution during test no. 2.

This time, however, in the case of the MPQUIC protocol, there was a definitely slower response to the drop in the available bandwidth as well as its growth. A similar situation takes place in the case of the MPTCP protocol. Its default configuration prefers a link with a lower RTT, in our case, Link no.1. As a result, in this test, a reduction in the value of the effectively used bandwidth was observed also in the case of this protocol. However, a system using MPTCP still offers better results.

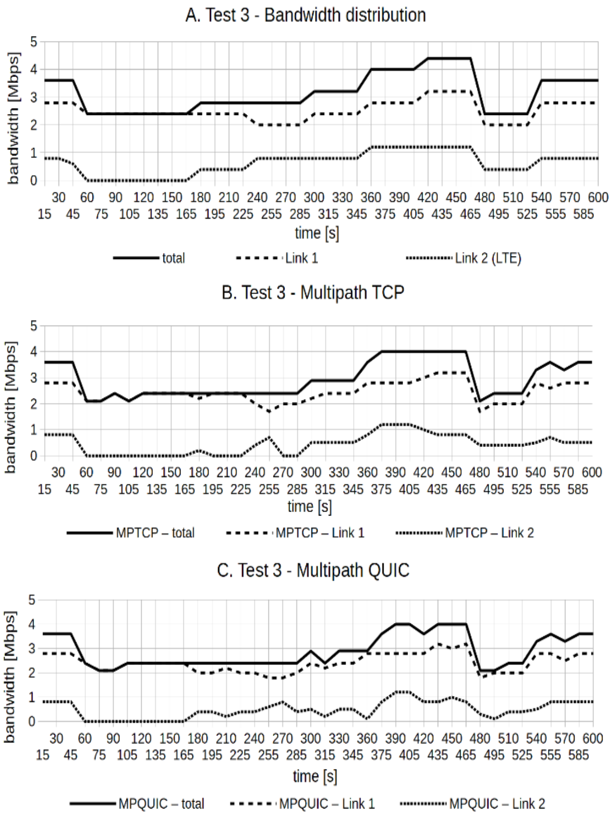

Fig. 4. Bandwidth distribution during test no. 3.

Considering the influence of the tested algorithms on the behaviour of the DASH adaptation process, the results coincide with those obtained in test no.1.

Table 4. Test no.2 – number of switches (average values).

Protocol	Number of switches		
	All	Down	Up
MPTCP	10	4	6
MPQUIC	13	5	8

However, attention should be paid (Table 4) to increase in the number of switches. The distribution of bandwidth between links in this test worsens the quality of the streaming video service compared to test no. 1. This is confirmed by the analysis of the PSNR value. In this case, the difference between the quality of the original video and received at the time of the highest degradation (switching to the lowest bit rate) was 18.2%, which is higher than that calculated for test no. 1. However, the decrease in the average PSNR value for the whole video (relative to the original quality) is only slightly higher and was equal to 7.26% compared to the value of 6.91% obtained for the test no. 1.

Table 5. Test no.2 – usage of available bandwidth.

Protocol	Bandwidth used					
	total		Link no. 1		Link no.2	
	avg.	dev	avg.	dev	avg.	dev
MPTCP	2.88	0.387	2.28	0.361	0.60	0.160
MPQUIC	2.75	0.319	2.19	0.322	0.57	0.114

The results, in terms of bandwidth used, which were obtained in the third test, are definitely better than in test no.2 and comparable with the first test. This confirms that both multipath transmission protocols perform relatively well in the situation of variable transmission conditions on both links. As can be seen in Figure 4, unfortunately, this feature has resulted in a distinct fluctuation of the bandwidth used in each of the links. This is also confirmed by the data contained in Table 7. Despite the equivalent average values of the total bandwidth in test no.1 and no.3, the values of the deviations in the test no.3 are definitely higher. This obviously affected the behaviour of the DASH service.

Table 6. Test no.3 – number of switches (average values).

Protocol	Number of switches		
	All	Down	Up
MPTCP	14	4	10
MPQUIC	17	6	11

In test no.3, the number of switches between individual video representations has increased. They were the most frequent among all tests carried out. In addition, an analogous increase occurred in the case of decisions about downloading chunk with lower parameters (switching down).

Table 7. Test no.3 – usage of available bandwidth.

Protocol	Bandwidth used					
	total		Link no. 1		Link no.2	
	avg.	dev	avg.	dev	avg.	dev
MPTCP	2.96	0.621	2.46	0.323	0.45	0.313
MPQUIC	2.92	0.589	2.43	0.327	0.49	0.298

This negatively affected the quality of the received video, which was confirmed by our subjective impressions during this test. The same conclusion is provided by the PSNR value analysis, which for this test showed the highest degradation of the quality of the video signal streamed. The decrease in the average PSNR value for the whole video (relative to the original quality) was the biggest because it was 11.22%. This test also brought the biggest change (reduction) in quality in the event of the client switching to receiving a video fragment with a lower bit rate. In the worst case of down switching, this decrease was 24.38%.

5 Conclusions

The growing availability of multihomed devices and simultaneous access to many network infrastructures (wired and wireless) highlights the need to consider the possibility of using multi-path data transmission. This can be particularly beneficial for all services in which a critical requirement is to provide a sufficient bandwidth on the data transmission path. One of the widely used services of this type is the video streaming service. In industrial networks, the bandwidth is typically shared among different services, often of a critical nature. Therefore, it is difficult to apply traffic policies based on unconditional preference for video broadcasts, *e.g.* as in the case of domestic networks. The introduction of the DASH / SAND standard has provided an opportunity to implement the principles of adaptation to services such as monitoring industrial processes or video surveillance. However, the low quality of streaming video that can occur when adapting to network overload conditions is still a problem. For this reason, the use of multipath transmission (and thus increasing the available bandwidth) is a highly promising direction of development.

Recent years have brought proposals for the implementation of multipath data transmission, which corresponds to the above assumptions. The article contains experiments based on two most popular solutions in this group, MPTCP and MPQUIC. Both protocols have been implemented in a typical streaming system based on DASH principles. Owing to these technologies, it was possible to summarise the properties of these protocols using their currently recommended (default) configuration.

Based on the results obtained, it can be concluded that the best conditions for streaming videos occur in a situation where the preferred transmission path is characterised by a stable available bandwidth. In the opposite case, there is a significant increase in the number of switches between video representations, which significantly affects the perceived quality of the video streaming service. In all the cases analysed, better results were found in the case of using multipath data transmission based on the MPTCP protocol.

References

1. ISO/IEC 23009-1:2014 (2014)
2. ISO/IEC FDIS 23009-5:2015(E) (2016)
3. T. Stockhammer, *Proc. ACM Conf. on Multimedia Systems* (ACM Press, 133-144, 2011)
4. S. Akhshabi, A. Begen, C. Dovrolis, Proc. ACM Conf. on Multimedia Systems, 157–168 (2011)
5. K. Khan, W. Goodridge, Int. Journal of Advanced Networking and Applications **9**, 3400–3410 (2017)
6. J. Iyengar, I. Swett, draft-tsvwg-quic-protocol-02, https://tools.ietf.org/html/draft-tsvwg-quic-protocol-02 (2016)
7. A. Ford, C. Raiciu, M. Handley, S. Barre, J. Iyengar, RFC **6182** (2011)
8. O. Bonaventure, C. Paasch, B. Detal, RFC **8041** (2017)
9. C. Raiciu, C. Paasch, S. Barre, Proc. USENIX Conf. On Networked Systems Design and Implementation, 29–29 (2012)
10. B. Biswal, O. Gnawali, *Proc. IEEE Global Communications Conference (GLOBECOM)* (IEEE, 1–6, 2016)
11. J. Iyengar, M. Thomson, draft-ietf-quic-transport-08, https://www.ietf.org/archive/id/draft-hamilton-quic-transport-protocol-01.txt (2017)
12. Q. De Coninck, O. Bonaventure, *Proc. Int. Conf. on Emerging Networking EXperiments and Technologies (CoNEXT)* (ACM Press, 160-166, 2017)
13. Q. De Coninck, O. Bonaventure, draft-deconinck-multipath-quic-00, https://tools.ietf.org/html/draft-deconinck-multipath-quic-00 (2017)
14. Q. De Coninck, M. Baerts, B. Hesmans, O. Bonaventure, Proc. Int. Passive and Active Measurements Conference, **17**, 57–69 (2016)
15. Y.-C. Chen, Y. Lim, R.J. Gibbens, E.M. Nahum, R. Khalili, D. Towsley, Proc. Internet Measurement Conference, 455–468 (2013)
16. C. James, E. Halepovic, M. Wang, R. Jana, N.K. Shankaranarayanan, *Proc. Int. Sym. on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)* (IEEE, 331–336, 2016)
17. B. Han, F. Qian, L. Ji, V. Gopalakrishnan*, Proc. Int. Conf. on Emerging Networking EXperiments and Technologies (CoNEXT)* (ACM Press, 129–143, 2016)
18. https://www.multipath-tcp.org/
19. https://multipath-quic.org/
20. C. Paasch, R. Khalili, O. Bonaventure, *Proc. Int. Conf. On Emerging Networking EXperiments and Technologies (CoNEXT)* (ACM Press, 393-398, 2013)
21. https://github.com/google/shaka-player
22. S. Lederer, C. Muller, C. Timmerer, *Proc. ACM Multimedia Systems Conference* (ACM Press, 22–24 (2012)