# Performance evaluation of video streaming using MPEG DASH, RTSP, and RTMP in mobile networks

A. Aloman, A.-I. Ispas, P. Ciotirnae
Department of Communications
Military Technical Academy
Bucharest, Romania
ciotirnae@mta.ro

R. Sanchez-Iborra, M.-D. Cano
Dept of Information Technologies and Communications
Universidad Politécnica de Cartagena
Cartagena, Spain
{ramon.sanchez, mdolores.cano}@upct.es

***Abstract*— Nearly three-fourths of the world's mobile data traffic will be video in the next 5 years. Video is one of the most demanding services in terms of network efficiency, reliability, and quality. In this work, we present a comparative performance evaluation of three different video streaming protocols, namely MPEG-DASH, RTSP, and RTMP for both on-demand and live video streaming over 4G and Wi-Fi (under different network conditions) in terms of Quality of user Experience (QoE). QoE measurements have been done (i) applying the recently standardized ITU-T Rec P-1201.1, which specifies the model algorithm for non-intrusive monitoring of video quality of IP-based video services based on packet header information for the lower resolution application area, and (ii) an extended non-standardized parametric model for comparative purposes. Results suggest that RTSP is more efficient than MPEG DASH for starting the video playback, but at the expense of decreasing QoE due to packet losses. We have also detected that PLR has a bigger influence over re-buffering events than end-to-end delay both in 4G/LTE and Wi-Fi, and that a slightly best quality is achieved by using QPSK at 20 MHz in 4G/LTE. QoE is noticeably higher with MPEG DASH than that attained by using RTSP, but slightly worse than that obtained with RTMP. Finally, our findings suggest that the use of parametric models for video QoE evaluation should be carefully review in terms of the weight that packet losses should have when streaming protocols based on reliable transport protocols (e.g., TCP) are used.**

***Keywords*—video streaming; Quality of user Experience; Quality of Service; Long Term Evolution; dynamic adaptive streaming; RTSP; RTMP; mobile wireless***

## I. INTRODUCTION

Within the next 5 years, monthly global mobile data traffic will surpass 24 exabytes, 4G traffic will be more than half of the total mobile traffic, and nearly three-fourths of the world's mobile data traffic will be video [1]. In all of its forms (video on demand, live streaming, conferencing, etc.), video is one of the most demanding services in terms of network efficiency and reliability, being for instance extremely low tolerant to packet loss and delay variations [2]. Therefore, fulfilling a high-quality video service becomes a must and a challenge for service providers and mobile network operators worldwide.

Quality measurements methods have notably evolved in telecommunication networks, from the classical Quality of Service (QoS) approach based on network parameters to the current efforts on Quality of user Experience (QoE), focused on assessing the level of quality the customer perceives when consuming a service. The most accurate methodology to assess QoE is performing subjective tests in which a panel of human testers rates the quality of the service under evaluation. This approach returns a quality score, so-called Mean Opinion Score (MOS), for the considered service in a scale 1 – 5; the higher the MOS the better the quality achieved. This methodology presents several drawbacks such as high expenses and high difficulties to perform tests at real time. For these reasons, the International Telecommunication Union (ITU) has recently released a new set of ITU-T Recommendations focused on parametric QoE monitoring of video services [3], [4].

In this work, we present the results of a comparative performance evaluation of video streaming protocols for both on-demand and live video streaming over 4G and Wi-Fi (IEEE 802.11g/n) in terms of QoE. Experiments have been carried out over real 4G and Wi-Fi networks under different network conditions by using the CMW500 equipment and smartphones (the latter either as video end-users or video content generators). We have tested the following video streaming protocols: MPEG Dynamic Adaptive Streaming over HTTP [5] and Real Time Streaming Protocol (RTSP)] [6] for 4G and Real Time Message Protocol (RTMP) [7] for Wi-Fi. QoE measurements have been done applying the recently standardized ITU-T Rec P-1201.1 [8], which specifies the model algorithm for non-intrusive monitoring of video quality of IP-based video services based on packet header information for the lower resolution application area (e.g., mobile devices). Our main contributions are (i) the comparative performance of adaptive and non-adaptive video streaming protocols under real network scenarios in terms of QoE, and (ii) the comparative evaluation of two parametric models for QoE evaluation and the effect that reliable transport transmission could have in the re-design of these planning and evaluation tools.

The rest of the paper is organized as follows. Related works are summarized in Section II. Section III includes a description of the testing scenario. Results are shown and

978-1-5090-0351-8/15 $31.00 © 2015 IEEE
DOI 10.1109/WMNC.2015.12

discussed in Section IV. Main conclusions are outlined in last section.

## II. Related work

There have been several studies on QoE for video services over LTE/4G based on simulation methodologies [9]–[16]. Similarly, several works can be found in the related literature addressing the benefits and cons of using adaptive or non-adaptive streaming protocols in mobile networks from the QoE perspective [9], [10], [16]–[20], and some works have studied the performance of live video streaming in QoE terms [21]–[24]. Nevertheless, only a few works have been carried out over real testbeds, e.g., [17], [25], and to the authors' knowledge none of them is based on the standard method recommended by ITU-T [3] for QoE measurements.

As stated in [25], there has been research on evaluating the advantages of different streaming techniques from the server performance perspective. For instance, it is well known that, compared with UDP-based streaming, adaptive streaming protocols simplify the rate adaptation process by providing multiple bitrate encodings for the same content [9]. However, even though most of these techniques are understood by the research community, a comprehensive study of these streaming techniques is still required from the perspective of the mobile device and the user. With our work we aim to work in this line, which will be further enriched in future works specifically focusing on the effect of these streaming techniques on user satisfaction.

### A. *Assessing video QoE in mobile networks: ITU-T Rec P.1201.1*

The ITU-T P.1201.1 model algorithm is a no-reference (non-intrusive) model that operates analyzing packet header information. Other input information not available from packet headers (e.g., video resolution) is provided to the model algorithm out-of-band. The model outputs individual estimates of audio, video, and audiovisual quality in terms of the five-point MOS scale. The main goal of this model is monitoring transmission quality during in-service operation or for maintenance purposes. ITU-T P.1201 is intended for UDP-based streaming and has been validated for encoding (compression) degradation of audio and video with a variety of bitrates, packet loss degradation, re-buffering degradation, and video contents of different spatio-temporal complexity. Likewise, the model produces reliable results with different video keyframe and frame rates (e.g., frame rates 5-30 Hz), different video resolutions (HVGA, QVGA, QCIF), different decoder-side packet loss concealment strategies, and different coding techniques (MPEG4 Part 2, ITU-T H.264).

However, the model (although applicable) needs further investigation for in-service monitoring of live network audiovisual and video TCP-based streaming. It is also important to note that this standard is expected to be extended in the future to include both adaptive and non-adaptive streaming over TCP, but currently no standard is available in these terms. Given the novelty of the standard, it has not been yet commonly used in the related literature as the reference model for video QoE evaluation in 4G/LTE/5G networks (as it does happens with other QoE evaluation models, e.g. ITU-T E-model for VoIP [26]). With our work, we provide experimental results on the applicability of the ITU-T Rec P.1201.1 for video streaming over different transport protocols.

### B. *Protocols for video streaming in mobile networks*

MPEG Dynamic Adaptive Streaming over HTTP (MPEG-DASH) [5] is a pull-based bitrate streaming standard where the client device plays the central role by providing the intelligence that drives the video adaptation. The basic idea is that content is made available to the client at a variety of different bit rates, so that the client can select the HTTP-based file with the highest bit rate possible that can be downloaded in time for play back without causing stalls or re-buffering events in the playback, i.e., the best option based on current network conditions. In contrast, Real Time Streaming Protocol (RTSP) [6] uses the server to keep track of the client state and to provide it with the streaming. On the other hand, the Real-Time Messaging Protocol (RTMP) [7] was designed for high-performance transmission of audio, video, and data between Adobe Flash Platform technologies, including Adobe Flash Player and Adobe AIR. Currently, RTMP is available as an open specification to create products and technology that enable delivery of video, audio, and data in the open AMF, SWF, FLV, and F4V formats compatible with Adobe Flash Player. Whereas MPEG-DASH and RTMP operate over a reliable stream transport, such as TCP, RTSP usually works over UDP.

## III. Test-bench

In this section, the experimental test-bench is described. The hardware configuration where the measurements have been carried out is illustrated in Fig. 1. The main element of a video-streaming transmission is the video server. In this case, the Wowza Streaming Engine [27] was installed in a laptop to perform this task. Wowza is a media streaming platform that provides live and on-demand video-streaming services to a wide variety of technologies, i.e., it can deliver multimedia content to many popular media players supporting many file formats. Regarding the network emulation, the wideband radio communication tester model Rohde & Schwarz CMW500 [28] was employed. This equipment is able to emulate different wireless networks and to perform IP measurements, as well as introducing controlled impairments into the communications. The smartphone Sony Xperia Z1 was employed as end-device. This phone includes a Qualcomm MSM89 Quad Core processor and 2 GB RAM, and runs Android 4.4.4. With this device, both video on-demand (smartphone as end device) and live video (smartphone as content generator) streaming services were tested during the study. Video on-demand was tested in a 4G/LTE network and live video streaming in a Wi-Fi 802.11g/n network, please see next sub-sections for further details.

### A. *Video on-demand over4G/LTE set-up*

In this scenario, the CMW500 base station was connected via Ethernet to the Wowza Video Server and via 4G/LTE to the mobile phone, which was used to consume the video on-demand service (please, see Fig. 1). LTE uses three types of modulation: QPSK, 16-QAM, and 64-QAM.

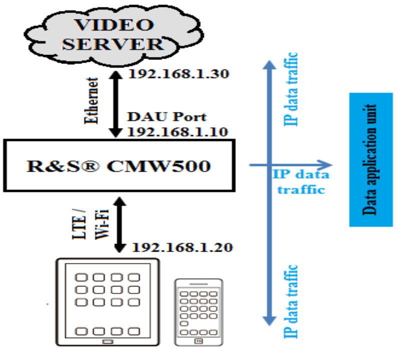



Fig. 1. Testing Scenario

For this study, two of the three types of modulation were employed, namely, QPSK and 64-QAM. Also, we carried out tests using two different channel bandwidths, 5MHz and 20 MHz.

Two different video-sources were employed for these tests, each of them with different coding bit-rates. The first video bit-rate was set to 570 Kbps and the second one was fixed to 1157 Kbps, both at 25 frames/second. Resolutions were 424 x 240 and 640 x 360, respectively. The first video consisted of a cartoon sequence and the second was extracted from a movie scene. Each clip lasted 2 minutes. Both sources were transmitted by using two different video-streaming technologies, namely, MPEG DASH based on TCP and RTSP that employs UDP. We employed codec H.264 and player "Movies" [29]. Using the capabilities of the R&S CMW500, different values of Packet Loss Rate (PLR) and delay were introduced, {0%, 0.5%, 1.5%, 3%, 6%} and {0 ms, 25 ms, 100 ms, 200 ms, and 400 ms}, respectively. Thus, we could observe these factors impact on video quality in terms of MOS.

### *B. Live video streaming over Wi-Fi set-up*

Using the R&S CMW500, an IEEE 802.11g/n Wi-Fi network was also configured. In this case, the smartphone was employed as video producer instead of as video consumer. This was achieved by using the Wowza GoCoder [30], which is a live audio and video encoding application for Apple and Android devices. It allows to encode live content right from the device and deliver it to the Wowza Streaming Engine in real time over 4G, 3G, and Wi-Fi systems. Then, the Wowza Streaming Engine can distribute this stream to video-clients. Thereby, the mobile phone was connected with the video server through the WLAN network created with R&S CMW500. GoCoder encoded the video using H.264 codec and transmitted the video-streaming to the server using the TCP-based RTMP. GoCoder offers the possibility of changing the encoding parameters. As in the previously described scenario, different configuration values were tuned aiming at analyzing their impact on the quality achieved by the live video service (please see Table I). Different levels of network impairments were also introduced in the transmissions. Each video-content created lasted 1 minute. The streams were played at the receiver using the Adobe player installed in the laptop.

### *C. Video quality estimation*

One of the aims of this work is investigating about the existing relationship between QoS and QoE. Although the measurement of QoS metrics, e.g., PLR, delay, jitter, etc., is quite straightforward, estimating the QoE in terms of MOS for a complex service such as video-streaming is not a trivial task. For that reason, different MOS estimators have been developed from different perspectives [31]. In this study, we have used the methodology presented in the ITU-T P.1201.1 [8], which specifies the algorithmic model for the lower resolution application area of ITU-T Rec P.1201 [3]. For both scenarios 4G/LTE and Wi-Fi, the complete transmission and reception packet-traces were captured by using tPacketCapture [32] app on the smartphone and Wireshark [33] in the laptop side.

TABLE I. CONFIGURATION AND IMPAIRMENTS FOR WI-FI SETUP

| Tested Parameters | Values |
|---|---|
| Resolution (px) | (320 x 240); (1280 x 720) |
| Coding bit-rate (Kbps) | 280 and 700 for (320 x 240)<br>1000 and 2500 for (1280 x 720) |
| Framerate (FPS) | 15; 30 |
| Key Frame Interval (s) | 2;10 |
| PLR (%) | {0; 0.5; 1; 2; 3} |
| Delay (ms) | {0; 50; 100; 150; 200} |

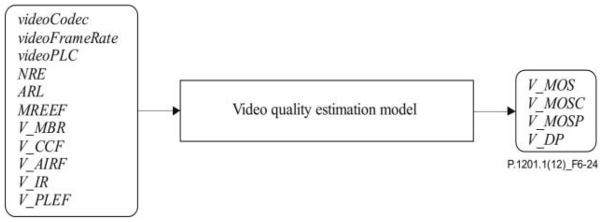



Fig. 2. Video quality estimation module (extracted from [8])

Although ITU-T Rec P.1201.1 model includes Audio Quality estimation module, Video Quality estimation module, and Audivisual Quality estimation Module, in this work it is only considered the video model to determine the MOS of the video streaming service as a first approach. The video quality estimation module is shown in Fig. 2. This MOS estimator was implemented in Matlab [34] by using the different elements described next. These elements permit determining the final video MOS depending on the intensity of different degradation types: (i) video quality due to compression (*V_MOSC*), (ii) video quality due to packet loss (*V_MOSP*), and (iii) video quality due to re-buffering (*V_MOSR*).

(i) Video quality due to compression (*V_MOSC*) is calculated as shown in (1), where, *MOS_MAX* represents the maximum reachable QoE (MOS) value, i.e., 5, and *V_DC* is the video distortion quality due to compression calculated as shown in (2). $MOS_{MIN}$ is the minimum QoE (MOS) value, i.e., 1; $v3$, $v4$, $v5$, and $v6$ are experimental coefficients depending on the coding scheme employed. *V_NBR* (Video Normalized video Bit-Rate) is the modified value of *V_BR* (Video Bit-Rate), in Kbps; it is calculated as shown in expression (3). *V_CCF* (Video Content Complexity Factor) is the factor describing the content's spatio-temporal complexity; its maximum value is 1, the initial value is 0.5, and it is calculated as shown in (4), where *V_ABIF* is the average number of bytes per I-frame.

$$V\_MOSC = MOS\_MAX - V\_DC \quad (1)$$

$$V_{DC} = \frac{MOS_{MAX} - MOS_{MIN}}{1+\left(\frac{V_{NBR}}{v3\cdot V_{CCF}+v4}\right)^{v5\cdot V_{CCF}+v6}} \quad (2)$$

$$V\_NBR = \frac{V\_BR\cdot 8\cdot 30}{1000\cdot \min(30, videoFrameRate)} \quad (3)$$

$$V\_CCF = min\left(\sqrt{\frac{V\_BR}{V\_ABIF\cdot 15}}, 1\right) \quad (4)$$

(ii) The video quality due to packet-loss ($V\_MOSP$) is calculated as described in (5). $V\_DP$ is the video distortion quality due to packet-loss. For additional details, please refer to ITU-T Rec. P.1201.1.

$$V\_MOSP = V\_MOSC - V\_DP \quad (5)$$

(iii) Video quality due to re-buffering ($V\_MOSR$) is calculated as shown in (6), where Video_Quality represents the initial video quality depending on the type of degradation situation involving re-buffering and $V\_DR$ is the video distortion quality due to re-buffering. For further details please refer to ITU-T Rec. P.1201.1 [8].

$$V\_MOSR = Video\_Quality - V\_DR \quad (6)$$

Thus, the final MOS is calculated as shown below:

$V\_MOS = V\_MOSC$ *(if no packet-loss and no re-buffering)*

$V\_MOS = V\_MOSP$ (if packet-loss and no re-buffering)

$V\_MOS = V\_MOSR$ (if packet-loss and re-buffering)

For comparison purposes, we will also employ the parametric model presented in [35] and the corrections to this model introduced in [36]. As shown in (7), this model allows obtaining accurate video-MOS estimations by means of PLR, the coding scheme (*k*), the resolution (*a*) and the video-coding bit-rate (*Br*) parameters.

$$V_q = 1 + 4k\left(1 - \frac{1}{1+\left(\frac{a\cdot Br}{v_1}\right)^{v_2}}\right)e^{-\frac{PLR}{v_3}} \quad (7)$$

In (7), *v1*, *v2*, and *v3* are experimental factors that depend on the coding scheme, the resolution applied, and the motion characteristics of the video. The values assigned to these factors were: *k*=1.12, *a*=10.8, *v1*=0.366, *v2*=1.32, and *v3*=3.5 respectively, consistent with its authors' recommendations.

# IV. Results

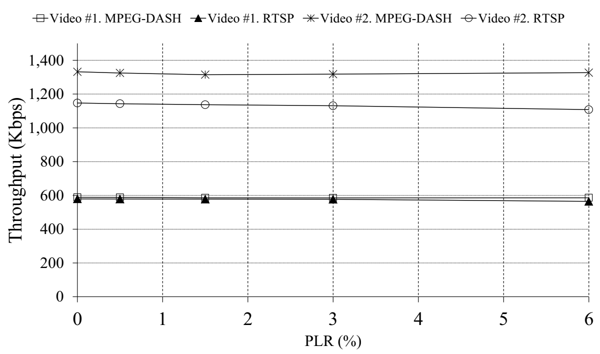
In this section, results are shown and discussed. Both experimental scenarios are described separately and common findings are summarized at the conclusions.

## A. Video on-demand over 4G/LTE: MPEG DASH and RTSP

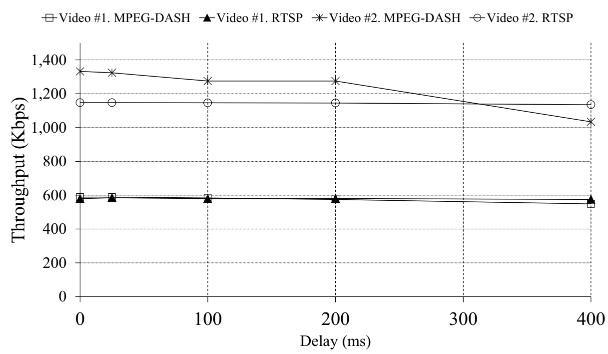
Firstly, the focus is on evaluating how the introduced impairments, namely, PLR and delay, impacted on a basic metric such as the throughput. Thus, Fig. 3 depicts the evolution of the throughput measured by the CMW500, for both video-sources (video #1: 570 Kbps; video #2: 1157 Kbps), when varying the intensity of the mentioned impairments. Both video-clips were transmitted by using the two streaming technologies mentioned above: an adaptive one (MPEG-DASH) and a non-adaptive approach (RTSP).

It can be observed a little decrease of the throughput value with the growth of both PLR and delay. This impact is slightly more noticeable in the case of MPEG DASH in the case of larger delays. We can observe in Fig. 3.b) how the throughput with MPEG DASH decreases as delay increases, whereas RTSP presents a more constant result. This behavior is due to the adaptive feature of MPEG DASH: when network degradation is detected, the transmitter reduces the transmission data rate to permit the receiver recovering from it. With RTSP, the throughput is more constant because this protocol does not try to attempt to retransmit damaged or delayed packets. Even so, the throughput achieved by MPEG DASH when transmitting the heavier video (video #2) is a little bit higher than that reached by RTSP.

Nevertheless, throughput stability shown by RTSP does not necessarily mean that the received video has a higher level of quality than the MPEG DASH transmission. (Re)buffering events could have a notable weight in the quality perceived by the end-user. For that reason, it has been also evaluated the initial buffering time and the number of re-buffering events occurred during the transmission. First, observe that MPEG DASH always requires longer initial buffering time (see Fig.4a and Fig. 4b) than RTSP. This is because UDP-based RTSP does not try to recover lost packets, so the protocol continues sending the video stream, although some prior packets could be lost. On the other hand, the TCP-based MPEG

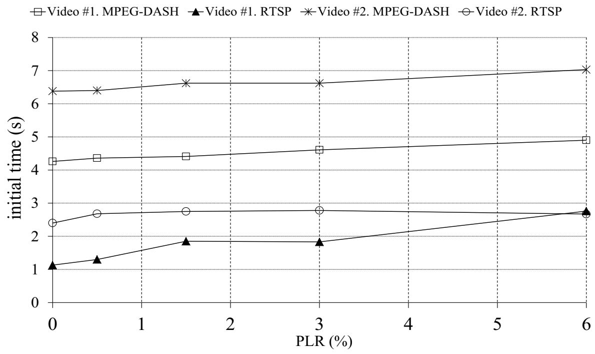



(a) Throughput (Kbps) vs PLR (%)

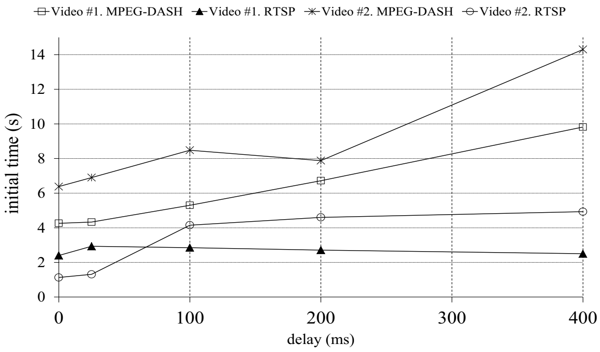



(b) Throughput (Kbps) vs delay (ms)

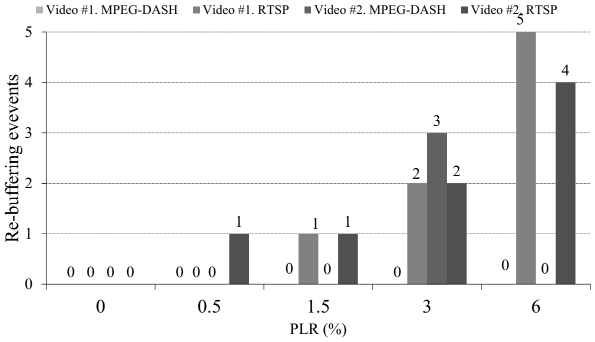
Fig. 3. Throughput achieved in the video-streaming communication for different levels of introduced PLR and delay. Results for two different video-sources delivered with two different streaming algorithms, MPEG DASH and RTSP.



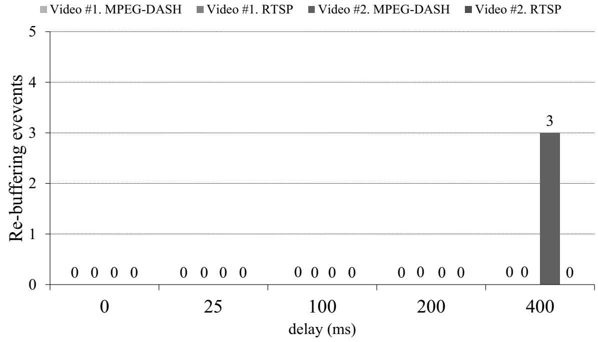

(a) Initial Time (s) vs PLR (%)




(b) Initial Time (s) vs delay (ms)




(c) Re-buffering events vs PLR (%)




(d) Re-buffering events vs delay (ms)

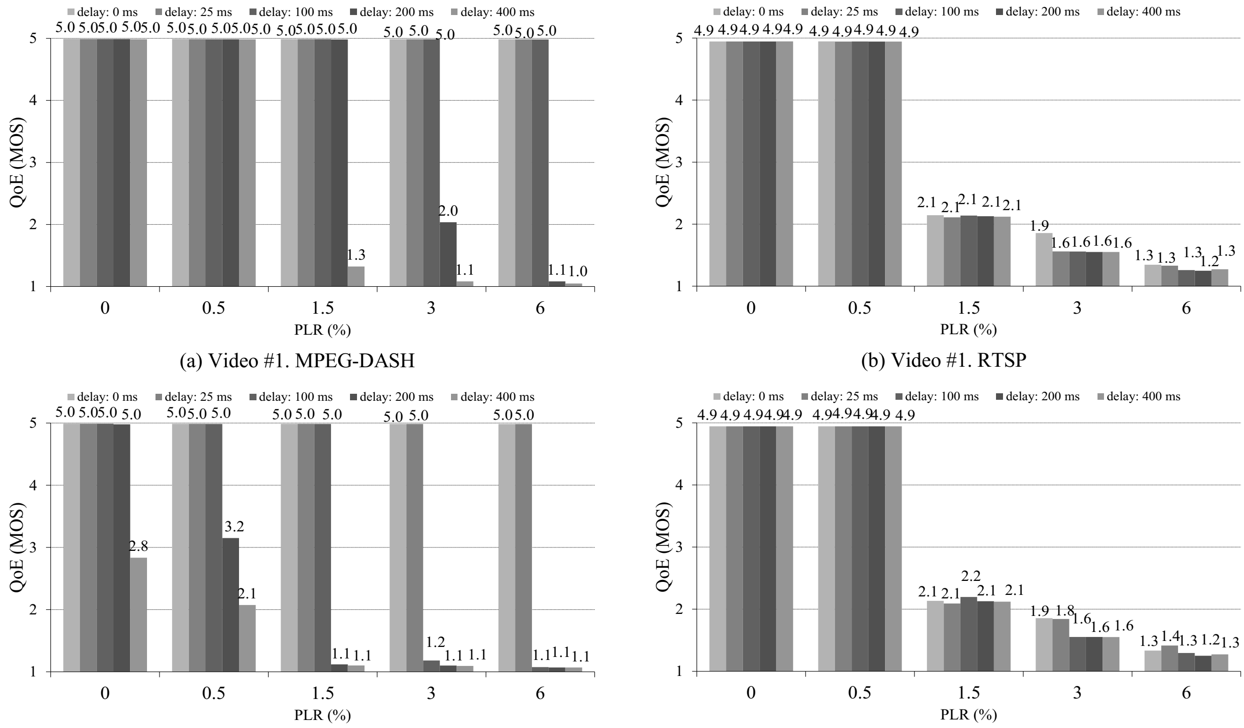
Fig.4. Impact of Packet Loss Rate and delay on initial buffering time and re-buffering events. Results for two different video-sources delivered with two different streaming algorithms, MPEG DASH and RTSP.

DASH algorithm re-sends lost packets until the packet-stream is complete in the receiver. These different approaches are also noticeable in the impact of PLR and delay on the pre-buffering time. Whereas RTSP does not seem to be highly affected by the introduced PLR or delay, MPEG DASH needs longer time to pre-fill the buffer when more packet losses or delay are introduced, especially in the case of delivering heavy traffic (video #2). Therefore, it can be concluded that RTSP is more efficient than MPEG DASH for starting the video playback, but at the expense of including more packet losses in the finally played video stream. This happens because RTSP is based in the UDP transport protocol. This protocol is low tolerant to packet loss due to its incapability to retransmit packets. Therefore, the loss of irrecoverable packets causes a deep decrease in the video quality, as discussed in next sections. In accordance to [25], the low quality video is played with a shorter initial delay than higher quality video.

However, the extra time needed by MPEG DASH for pre-buffering the video sequence is not in vain. Observe in Fig. 4(c) and Fig. 4(d) how the number of re-buffering events, which are considered highly annoying for the end-users, is much greater for the case of RTSP. In other words, the long pre-loading time interval needed by MPEG DASH permits to alleviate the impact of the packet losses happened during the transmission, revealed by a null number of re-buffering events in all scenarios but one. Furthermore, observe how PLR has a bigger influence over these events than the end-to-end delay. This is because the video-buffer is able to successfully reduce the impact of the delayed packets.

After evaluating the system from a QoS perspective, results are now analyzed from the end-user point of view (QoE). Fig. 5 presents the QoE (MOS) obtained when video #1 is transmitted under the same PLR/delay conditions described above. Observe how in almost all scenarios, the QoE (MOS) achieved by using the MPEG DASH protocol is noticeably higher than that attained by using RTSP. As explained before, for the case of the RTSP protocol, the network impairments do not affect the initial buffering time or re-buffering events, but the image quality suffers great quality degradation due to the influence of packet loss. In turn, because of the adaptive nature of MPEG DASH and its longer pre-buffering time, the video playback is much smoother and the image quality is greater than that obtained by using RTSP. This is evidenced by the high QoE (MOS) values reached in all evaluated scenarios (except the case of the video #2 with 400 ms of introduced delay). In Fig. 5(b) it can be seen that delay does not have such a strong impact on the QoE (MOS) as packet loss has (Fig. 5(a)), especially in the case of RTSP.

Additionally, given that in this experiment the connection between the mobile phone and the CMW500 was implemented using 4G/LTE, two parameters regarding this link configuration have been also evaluated, namely, channel bandwidth and modulation. Two different values for channel bandwidth are considered, namely, 5 MHz and 20 MHz. Regarding modulation, the experimental tests were carried out using QPSK and 64-QAM. Fig. 6 illustrates the QoE, in terms of MOS, attained for the video #1 and video #2 when introducing in the streams a delay of 100 ms and PLR of 1.5%. These




(a) Video #1. MPEG-DASH

(b) Video #1. RTSP

(c) Video #2. MPEG-DASH

(d) Video #2. RTSP

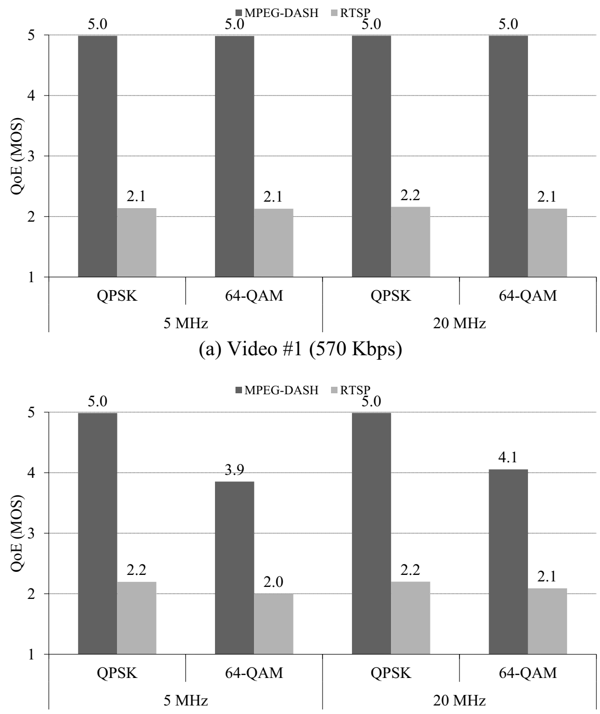
Fig.5. QoE (MOS) for different PLR and delays. Results for two video-sources delivered with two different streaming algorithms, MPEG DASH and RTSP.

figures have been selected to evaluate the impact of the LTE physical layer at the maximum recommendable values for the considered impairments (please, see ITU-T Rec. G.114 and G.1010). Observe the overall QoE (MOS) reduction when employing 64 QAM instead of QPSK. This is especially noticeable in the case of the video #2 streamed with MPEG DASH (Fig. 6(b)). Regarding the channel bandwidth, althougha single video transmission takes up very little of channel capacity, the video quality is slightly greater in the case of the wider bandwidth (please, observe the results for both videos streamed with RTSP).

Although not shown here due to the limited space, QoE (MOS) values obtained with the parametric model described in (7) are below those obtained with ITU-T Rec P.1201.1. These preliminary findings suggest that the effect of packet losses on parametric models should be carefully taken into account when streaming protocols used over reliable transport protocols, e.g., TCP, are employed.

### B. Live video streaming over Wi-Fi

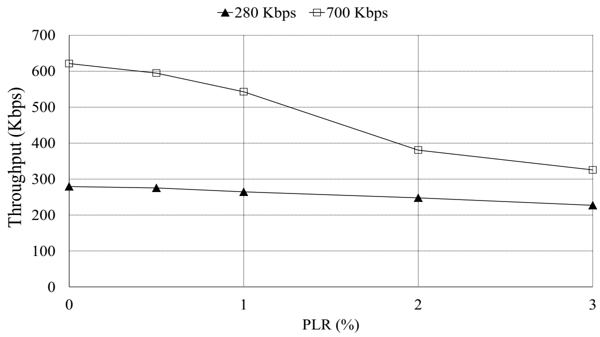
In this set-up the experiment perspective was changed. The smartphone acted as video transmitter, instead of as video-consumer. By using the Wowza GoCoder, different video-sequences of 1 minute were encoded by the mobile phone and afterwards streamed at real time through the Wowza Streaming Engine. Video sequences were recorded in QVGA resolution at 280 Kbps and 700 Kbps. Recall that the TCP-based RTMP protocol was employed in this case for the streaming transmission. Similar to the previous test-bench, different QoS metrics were evaluated (PLR and delay). Thus, Fig. 7 depicts the video streams throughput evolution with different values of introduced PLR (Fig. 7(a)) and delay (Fig. 7(b)). As expected, observe how both impairments negatively affect the throughput achieved by the transmission. As it happened with MPEG DASH, this impact is more noticeable in the case of the more demanding video stream (700 Kbps). On the other hand, the lower bit –rate transmission seems to be more robust to



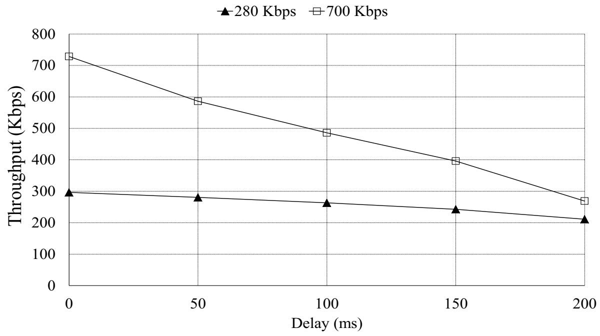

(a) Video #1 (570 Kbps)

(b) Video #2 (1157 Kbps)

Fig.6. QoE (MOS) comparison for different channel-bandwidths and modulations. Results for two different video-sources delivered with two different streaming algorithms, MPEG DASH and RTSP.




(a) Throughput (Kbps) vs PLR (%)




(b) Throughput (Kbps) vs delay (ms)

Fig. 7. Throughput achieved in the QVGA video streaming for different levels of introduced PLR and delay. Results for two different coding bit rates.

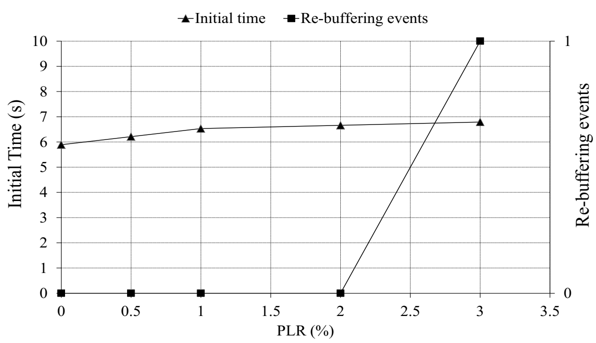



Fig.8. Impact of PLR on initial buffering time and re-buffering events. Results for QVGA video stream at 280 Kbps.

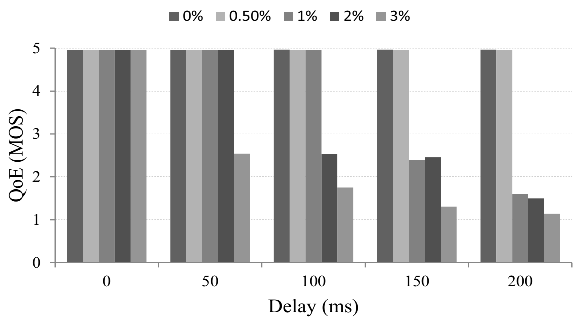



Fig.9. QoE evolution in terms of MOS with different values of introduced PLR(%) and delay (ms). Results for QVGA video stream at 280 Kbps.

the impact of the impairments. Regarding the effect of PLR on the buffering events, Fig. 8 shows how the packet loss has little impact on the pre-loading time or the number of re-buffering events, behaving quite similar to MPEG DASH protocol.

From a QoE perspective, Fig. 9 shows the QoE (MOS) evolution for the QVGA transmission (at 280 Kbps) for variable levels of PLR and delay. Observe that PLR has a greater impact on the perceived quality than the delay. This is because of the effect of the buffer, which permits to mitigate the negatively effect of delay or jitter. However, with the greatest values of PLR (2% and 3%), the QoE (MOS) suffers a pronounced drop with moderate levels of delay. In comparison with MPEG DASH and RTSP, RTMP remains faintly above MPEG DASH and notable above RTSP in terms of overall achieved MOS.

## V. Conclusion

We have conducted a comparative performance evaluation of MPEG DASH, RTSP, and RTMP streaming protocols over 4G and Wi-Fi (IEEE 802.11g/n) real networks in terms of QoE. We have tested both video on demand and live video streaming. Results suggest that RTSP is more efficient than MPEG DASH for starting the video playback, but at the expense of decreasing QoE due to packet losses. In addition, the long pre-loading time interval needed by MPEG DASH or RTMP permits to alleviate the impact of the packet losses happened during the transmission, as revealed by a lower number of re-buffering events for these two protocols. We have also detected that PLR has a bigger influence over these events than end-to-end delay both in 4G/LTE and Wi-Fi, and that a slightly best quality is achieved by using QPSK at 20 MHz in 4G/LTE. QoE is noticeably higher with MPEG DASH than that attained by using RTSP, but slightly worse than that obtained with RTMP. Finally, our findings suggest that the use of parametric models for video QoE evaluation should be review in terms of the weight that packet losses should have when streaming protocols based on reliable transport protocols (TCP) are used. Our future work will widen the comparative results found here, further enriching them by focusing on the effect of these streaming techniques on user satisfaction.

## Acknowledgment


This work was supported by the MINECO/FEDER project grant TEC2013-47016-C2-2-R (COINS).


## References


[1] Cisco, “Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2014–2019. White Paper,” *White Pap.*, 2015.

[2] P. Rengaraju, C.-H. Lung, F. Yu, and A. Srinivasan, “On QoE monitoring and E2E service assurance in 4G wireless networks,” *IEEE Wirel. Commun.*, vol. 19, no. 4, pp. 89–96, Aug. 2012.

[3] “Parametric non-intrusive assessment of audiovisual media streaming quality,” *ITU-T Recomm. P.1201*, 2012.

[4] “Parametric non-intrusive bitstream assessment of video media streaming quality,” *ITU-T Recomm. P.1202*, 2012.

[5] “ISO/IEC 23009-1:2014 ‘Information technology -- Dynamic adaptive streaming over HTTP (DASH) -- Part 1: Media presentation description and segment formats,’” 2014.

[6] “Real-Time Streaming Protocol (RTSP),” *IETF RFC 2326*, 1998.

[7] M. T. H. Parmar, “Adobe’s Real Time Messaging Protocol,” *Adobe Syst. Inc.*, 2012.

[8] "Parametric non-intrusive assessment of audiovisual media streaming quality - lower resolution application area," *ITU-T Recomm. P.1201.1*, 2012.

[9] A. El Essaili, D. Schroeder, E. Steinbach, D. Staehle, and M. Shehada, "QoE-based traffic and resource management for adaptive HTTP video delivery in LTE," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 6, pp. 988–1001, Jun. 2015.

[10] J. Navarro-Ortiz, P. Ameigeiras, J. M. Lopez-Soler, J. Lorca-Hernando, Q. Perez-Tarrero, and R. Garcia-Perez, "A QoE-aware scheduler for HTTP progressive video in OFDMA systems," *IEEE Commun. Lett.*, vol. 17, no. 4, pp. 677–680, Apr. 2013.

[11] S. Singh, O. Oyman, A. Papathanassiou, D. Chatterjee, and J. G. Andrews, "Video capacity and QoE enhancements over LTE," in *IEEE International Conference on Communications (ICC)*, 2012.

[12] M. Shehada, S. Thakolsri, and W. Kellerer, "QoE-based resource reservation for unperceivable video quality fluctuation during Handover in LTE," in *IEEE 10th Consumer Communications and Networking Conference (CCNC)*, 2013, pp. 171–177.

[13] R. Perera, A. Fernando, T. Mallikarachchi, H. K. Arachchi, and M. Pourazad, "QoE aware resource allocation for video communications over LTE based mobile networks," in *10th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, 2014, pp. 63–69.

[14] V. Ramamurthi, O. Oyman, and J. Foerster, "Video-QoE aware resource management at network core," in *IEEE Global Communications Conference*, 2014, pp. 1418–1423.

[15] E. Yaacoub and Z. Dawy, "Network QoE metrics for assessing system-level performance of radio resource management algorithms in LTE networks," in *International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2014, pp. 411–416.

[16] O. Oyman and S. Singh, "Quality of experience for HTTP adaptive streaming services," *IEEE Commun. Mag.*, vol. 50, no. 4, pp. 20–27, Apr. 2012.

[17] J. De Vriendt, D. De Vleeschauwer, and D. C. Robinson, "QoE model for video delivered over an LTE network using HTTP adaptive streaming," *Bell Labs Tech. J.*, vol. 18, no. 4, pp. 45–62, Mar. 2014.

[18] N. Staelens, J. De Meulenaere, M. Claeys, G. Van Wallendael, W. Van den Broeck, J. De Cock, R. Van de Walle, P. Demeester, and F. De Turck, "Subjective quality assessment of longer duration video sequences delivered over HTTP adaptive streaming to tablet devices," *IEEE Trans. Broadcast.*, vol. 60, no. 4, pp. 707–714, Dec. 2014.

[19] M. Claeys, S. Latre, J. Famaey, and F. De Turck, "Design and evaluation of a self-learning HTTP adaptive video streaming client," *IEEE Commun. Lett.*, vol. 18, no. 4, pp. 716–719, Apr. 2014.

[20] Y. Xu, Y. Zhou, and D.-M. Chiu, "Analytical QoE models for bit-rate switching in dynamic adaptive streaming systems," *IEEE Trans. Mob. Comput.*, vol. 13, no. 12, pp. 2734–2748, Dec. 2014.

[21] G. Xi, X. Zhang, and K. Q. Beijing, "On adaptive live streaming in mobile cloud computing environments with D2D cooperation," in *21st International Conference on Telecommunications (ICT)*, 2014, pp. 405–409.

[22] K.-C. Fung and Y.-K. Kwok, "A QoE based performance study of mobile peer-to-peer live video streaming," in *13th International Conference on Parallel and Distributed Computing, Applications and Technologies*, 2012, pp. 707–712.

[23] A. Gouta, C. Hong, D. Hong, A.-M. Kermarrec, and Y. Lelouedec, "Large scale analysis of HTTP Adaptive Streaming in mobile networks," in *IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, 2013, pp. 1–10.

[24] A. El Essaili, L. Zhou, D. Schroeder, E. Steinbach, and W. Kellerer, "QoE-driven live and on-demand LTE uplink video transmission," in *IEEE 13th International Workshop on Multimedia Signal Processing*, 2011, pp. 1–6.

[25] M. Ashraful Hoque, M. Siekkinen, J. K. Nurminen, M. Aalto, and S. Tarkoma, "Mobile Multimedia Streaming Techniques: QoE and Energy Consumption Perspective," *Pervasive Mob. Comput. J.*, vol. May, 2014.

[26] "The E-model: a computational model for use in transmission planning," *ITU-T Recomm. G.107*, 2011.

[27] "Wowza Streaming Engine," 2015. [Online]. Available: http://www.wowza.com/products/streaming-engine.

[28] "R&S CMW500 - Platform Overview," 2015. [Online]. Available: http://www.rohde-schwarz.com/en/product/cmw500_overview-productstartpage_63493-10844.html.

[29] "Movies app," 2015. [Online]. Available: http://www.sonymobile.com/in/apps-services/sony-media-apps/.

[30] "Wowza GoCoder," 2015. [Online]. Available: http://www.wowza.com/products/gocoder.

[31] S. Chikkerur, V. Sundaram, M. Reisslein, and L. J. Karam, "Objective video quality assessment methods: a classification, review, and performance comparison," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 165–182, Jun. 2011.

[32] "tPacketCapture," 2015. [Online]. Available: http://www.taosoftware.co.jp/en/android/packetcapture/

[33] "Wireshark," 2015. [Online]. Available: http://www.wireshark.org.

[34] "Matlab," 2015. [Online]. Available: http://www.mathworks.es/products/matlab/.

[35] K. Yamagishi and T. Hayashi, "Parametric packet-layer model for monitoring video quality of IPTV services," in *IEEE International Conference on Communications*, 2008, pp. 110–114.

[36] J. Joskowicz and J. C. L. Ardao, "Enhancements to the opinion model for video-telephony applications," in *5th International Latin American Networking Conference-LANC '09*, 2009, p. 87.