

Empirical evaluation of H.265/HEVC based dynamic adaptive video streaming over HTTP (HEVC-DASH)

Iheanyi Irondi^{*a}, Qi Wang^a, Christos Grecos^a

^aCentre for Audio Visual Communications and Networks, School of Computing, University of the West of Scotland, Paisley PA1 2BE, United Kingdom.

ABSTRACT

Real-time HTTP streaming has gained global popularity for delivering video content over Internet. In particular, the recent MPEG-DASH (Dynamic Adaptive Streaming over HTTP) standard enables on-demand, live, and adaptive Internet streaming in response to network bandwidth fluctuations. Meanwhile, emerging is the new-generation video coding standard, H.265/HEVC (High Efficiency Video Coding) promises to reduce the bandwidth requirement by 50% at the same video quality when compared with the current H.264/AVC standard. However, little existing work has addressed the integration of the DASH and HEVC standards, let alone empirical performance evaluation of such systems.

This paper presents an experimental HEVC-DASH system, which is a pull-based adaptive streaming solution that delivers HEVC-coded video content through conventional HTTP servers where the client switches to its desired quality, resolution or bitrate based on the available network bandwidth. Previous studies in DASH have focused on H.264/AVC, whereas we present an empirical evaluation of the HEVC-DASH system by implementing a real-world test bed, which consists of an Apache HTTP Server with GPAC, an MP4Client (GPAC) with openHEVC-based DASH client and a NETEM box in the middle emulating different network conditions. We investigate and analyze the performance of HEVC-DASH by exploring the impact of various network conditions such as packet loss, bandwidth and delay on video quality. Furthermore, we compare the Intra and Random Access profiles of HEVC coding with the Intra profile of H.264/AVC when the correspondingly encoded video is streamed with DASH. Finally, we explore the correlation among the quality metrics and network conditions, and empirically establish under which conditions the different codecs can provide satisfactory performance.

Keywords: H.265/HEVC, video streaming, DASH, performance evaluation.

1. INTRODUCTION

Recent years have witnessed a remarkable increase in the demand for multimedia applications in Internet. A recent survey by Cisco¹⁸ indicates that over 69% of mobile traffic data will be generated by mobile video in 2018. The sharing of rich media data over the Internet comes with a lot of challenges such as the limited availability of the bandwidth and other time-varying network conditions in terms of packet loss, delay and jitter. There are several techniques and protocols for streaming video⁶ including traditional streaming, progressive download and more recently adaptive streaming, which addresses the limitations of the previous streaming techniques. MPEG-DASH (Dynamic Adaptive Streaming over HTTP) is an ISO/IEC adaptive streaming standard that employs client-server architecture, where the streaming session is controlled by the client and the streaming video is adapted to the client's network conditions¹⁰. In the DASH standard, a video stream is encoded into several bitrates and segmented into chunks, which are then stored on the HTTP web server. The system runs on the popular HTTP over TCP and as such does not need specialized servers and also the streams can easily traverse firewalls, making the system more attractive when compared with alternative, conventional solutions based on the Real-time Transport Protocol (RTP) over UDP⁶.

^{*}iheanyi.irondi@uws.ac.uk

Considering the limitations in bandwidth for streaming applications and the emergence of next-generation Ultra HD videos (4K, 8K etc.) with high bandwidth requirements, a new video coding standard H.265/HEVC has been developed and recently standardized, which is able to reduce the bandwidth requirement by 50% when compared with the current standard H.264/AVC. HEVC research is relatively new and most of existing research effort has centered on the design and evaluation of new and improved coding algorithms and tools. In contrast, transmission of HEVC encoded video over Internet is an emerging research area.

Very few existing projects have addressed HEVC networking systems, and most of the proposed/evaluated systems were based on RTP over UDP. For instance, Nightingale, Wang and Grecos proposed and implemented a physical testbed and software framework, referred to as HEVStream¹⁹, for evaluating HEVC streaming using RTP over UDP in lossy IP networks. They further designed and implemented a method for prioritizing HEVC packets to reduce the impact of packet loss²¹. Ozta et al.²⁰ investigated the performance of HEVC video streaming, again using RTP over UDP, in a mobile environment through simulations.

Regarding DASH, most previous studies have been limited to using the H.264/AVC based codecs. For example, Lederer et al. examined the impact of delay on H.264/AVC based DASH system¹⁰. The impact of different video segment sizes on the performance of H.264/AVC based DASH system was also investigated. Sanchez et al.²² further proposed an enhanced DASH system by employing H.264/SVC (Scalable Video Coding), which is the scalable extension to H.264/AVC. The integration and performance of H.265/HEVC based DASH system is an area worth investigating; however, little related work has been reported in the literature.

In this paper, we focus on investigating the use of HEVC with DASH under various Internet (Wide Area Network or WAN) conditions e.g., in loss prone networks and comparing the performance with the H.264/AVC based DASH system. We establish a physical test bed and empirically evaluate the performance of H.265/HEVC DASH system by measuring the video quality with the widely adopted PSNR metric. We study the impact of various WAN characteristics by introducing different values of delay and packet loss on H.265/HEVC Intra profile and Random Access profile systems and compare them with H.264/AVC based systems. We also study the impact of different video segment sizes on the HEVC DASH system by segmenting the test video sequence into 2-second and 10-second segments depicting a short and long segment size respectively, and then compare the effect of the different segment sizes on the different codecs. The reminder of the paper is organized as follows. Section 2 provides an overview of HEVC, DASH and related work. Section 3 describes the proposed test bed, the conducted experiments and the experimental results. Section 4 provides the conclusion.

2. RELATED WORK

2.1 H.265/HEVC (High Efficiency Video Coding)

H.265/HEVC is the latest video compression standard developed by ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Picture Experts Group (MPEG). The H.265/HEVC is the next-generation video standard succeeding the H.264/AVC, and in various independent tests it has been empirically proved to be able to reduce the bandwidth requirement by about 50% at the same video quality when compared with H.264/AVC¹. The HEVC standard can support 8K and up to 8192×4320 resolution Ultra HD videos to address the increasing demands and high bandwidth requirements of Ultra HD storage and delivery².

H.265/HEVC is a hybrid codec similar to H.264/AVC and has been designed to increase the use for parallel processing and increased video resolution³. A significant difference between H.264/AVC and H.265/HEVC lies in the frame coding structure². The core coding layer of the H.264/AVC is the microblock consisting of 16 × 16 block of luma samples and 8 × 8 Chroma samples while the H.265/HEVC has a quad tree like structure where each frame is divided into Coding Tree Units (CTU) which is then split into Chroma and luma Coding blocks (CB). A combination of one luma CB and two Chroma CBs and associated syntax form the Coding Unit (CU) where the CU can further be split into Prediction Units (PU) and Transform Units (TU)³.

The H.265/HEVC has a deblocking filter (DB) similar to the one in H.264/AVC¹ but is designed to be more suitable for parallel processing². A new feature called Sample Adaptive Offset (SAO) filter was also introduced in the decoding loop of the H.265/HEVC which provides better reconstruction of the original signal amplitudes. However, this feature introduces some form of complexity in the system. Slices are used to support parallel processing in H.264/AVC even

though they are more designed for error resilience and was also retained in H.265/HEVC. Other additional features added to H.265/HEVC include tiles and wavefront and in order for a decoder to perform parallel processing of tiles and wavefront, the position where each tile and slice starts in the bitstream must be determined¹.

2.2 Traditional Streaming and Progressive Download

Delivery of multimedia content across the Internet can be achieved by traditional streaming, progressive download and adaptive video streaming⁵. Traditional streaming is usually based on stateful push-based protocols, where the streaming server keeps track of the client's state (through PLAY, PAUSE etc. commands from the client) during the session and the RTP-based video packets are pushed towards the client. Real-Time Streaming Protocol (RTSP), a protocol for controlling media servers, is a typical example of traditional streaming scheme, which transmits packet using either UDP or TCP. The UDP transport protocol however has a high blocking probability when traversing firewalls and proxies⁷ and thus TCP may be preferred.

Progressive download⁵ is simply described as downloading files from an HTTP web server to the client while allowing the client to play the file when the downloading operation is still in progress. However, the progressive download has a few drawbacks because the bitrate has to be selected before the downloading and remains constant throughout the operation.

Adaptive video streaming is an improvement over traditional streaming and progressive download and the operation relies on HTTP. It works by detecting the client's network condition and CPU capacity in real time and adjusts the quality of the video stream. HTTP is a stateless pull-based protocol, and runs over TCP, which offers a number of advantages over UDP. In addition to those that have been mentioned, the Internet is built on HTTP where most network nodes support HTTP, thereby eliminating the need for specialized servers. On the other hand, TCP can result in added overhead and delays due to packet retransmission, which would cause video buffering/re-buffering issues.

2.3 MPEG-DASH

DASH is a new ISO/IEC MPEG adaptive streaming solution that enables high-quality media delivery over the Internet in a highly adaptive and dynamic approach with the aid of pervasive HTTP web servers⁶. This technique is designed to address the shortcomings of the progressive download and RTSP-based streaming and is suited for cases where there is constant variation in bandwidth or changing network conditions. DASH also allows the reuse of existing Content Delivery Network (CDN) infrastructure. It is noted that different industry proprietary solutions have been proposed and developed by Adobe⁶, Apple⁶ and Microsoft⁵, adopting a similar approach in their design.

A DASH system employs a client-server based architecture, where the media contents will be transported on top of the Internet infrastructure from the server to the client with the client fully controlling the streaming session. The system is made up of an HTTP web server, a DASH client, and the Media Presentation Description (MPD), which is an XML document that describes the media available to the server⁴. In DASH, the media files are encoded at different bitrates, resolutions, etc and chopped into segments and stored on the web server where the client requests for the MPD file using the standard HTTP GET requests before it obtains the segments that suits its capability⁷, thereby ensuring that the streaming is dynamically adapted to its bandwidth constraint⁴. An example DASH operation is illustrated in Figure 1. As shown in Figure 1, there are four representations of a video encoded at various bitrates for a particular period; each representation is then segmented into four chunks. During the DASH streaming session, the DASH client requests for the representation or chunk that suits its network condition where it dynamically adapts to other chunks (quality switching) in response to its network condition changes.

Adaptive streaming over HTTP based on the DASH standard is an ongoing research area and has attracted a lot of attention in the research community. In the rest of this section, we review some of the existing studies. DASH application in a mobile environment was proposed and carried out under real-world conditions⁶ and compared with different industry proprietary solutions by Adobe, Apple⁹ and Microsoft and the results showed that the proposed system can compete with the industry solutions.

Moreover, the segment size is an interesting area of research and it has been shown to have an impact on the performance of the DASH system, where shorter segments can result in large overhead¹⁰ in terms of requests and can also provide more quality switches, while larger segments are not very friendly in high bandwidth fluctuation environments like mobile networks because the client cannot adjust the video quickly in case of a significant drop in bandwidth. A segment-based DASH system from a teletraffic point of view was proposed² and the results showed that small segment sizes increase the probability of empty buffer and active buffer overflow while large segment sizes

decrease the probability of empty buffer and active buffer. Moreover, Lederer et al. developed a content generation tool for DASH called DASHEncoder¹⁰. The impact of different segment lengths on different HTTP settings and also the impact of delay and varying bandwidth on the network were evaluated by measuring the corresponding PSNR values, and their result demonstrated the advantage of persistent connection over non-persistent connection. However, their experiment was conducted using H.264/AVC and did not consider the impact of packet loss.

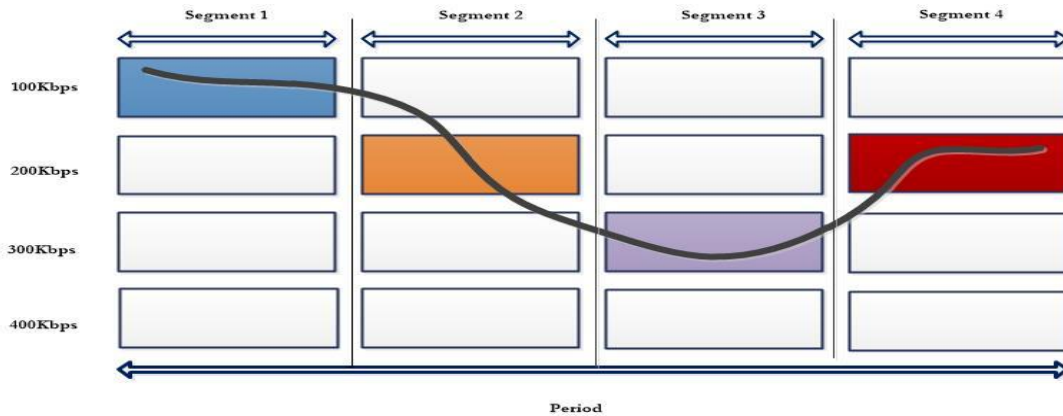


Figure 1. Example DASH operation: quality switching

Furthermore, a medical video content based DASH system was evaluated¹¹ by proposing adaptation approaches which showed that the quality of video streaming can be improved by selecting a suitable representation or segment size with an adaptation approach, a larger segment size with margin based approach improves the performance of ultrasound systems while smaller segment size with no-margin approach increases the performance of other streaming application systems, however this evaluation did not consider the impact of different codecs and network conditions.

In addition, the effect of network path quality on the Quality of Experience (QoE) of HTTP video streaming was studied¹³. The results showed that RTT and packet loss lowers the network throughput whilst increasing the re-buffering frequency which was the main factor responsible for the variations in the QoE scores through subjective evaluation. A novel estimation method for connection throughput was proposed¹⁴ by Thang et al. and it demonstrated stability to short term fluctuations while responding rapidly to large fluctuations. Bandwidth consumption of DASH applications was studied¹⁵ and the impact of different implementations of DASH and various access networks on bandwidth consumption were demonstrated. The use of multipath transmission to improve the average quality of adaptive HTTP streaming when compared with individual path was proposed and demonstrated¹⁶.

Finally, DASH integration in HEVC was discussed⁴ and the paper highlighted the advantage of HEVC over H.264/AVC for DASH implementation. Nevertheless, their work did not carry out any practical demonstration of the H.265/HEVC bitstream transmission in a DASH system. A combination of HEVC and DASH was used to propose a Power HEVC Streaming System for mobile environments¹⁷, and it demonstrated the ability to allow clients to switch adaptively among bitstream segments with varying complexity levels.

3. EMPIRICAL EVALUATION

3.1 Test Bed

This section describes the test bed used for the DASH experiment as illustrated in Figure 2. It comprises a web server and DASH streamer, a network emulator emulating different WAN characteristics and a DASH client. The server is a standard Apache web server running on a PC with Intel Core i5-2400 CPU @ 3.10GHz × 4, 500GB HDD, and Ubuntu 12.10 Linux operating system. The WAN emulator is a NETEM-based system running on a second PC with an Intel Core i3-2100 CPU @ 3.10GHz × 4, 500GB HDD, and Ubuntu 12.04 operating system, whilst the DASH client is a GPAC MP4Client based system running on a third PC with an Intel Core i3-2100 CPU @ 3.10GHz × 4, 500GB HDD, and Ubuntu 12.04 operating system.

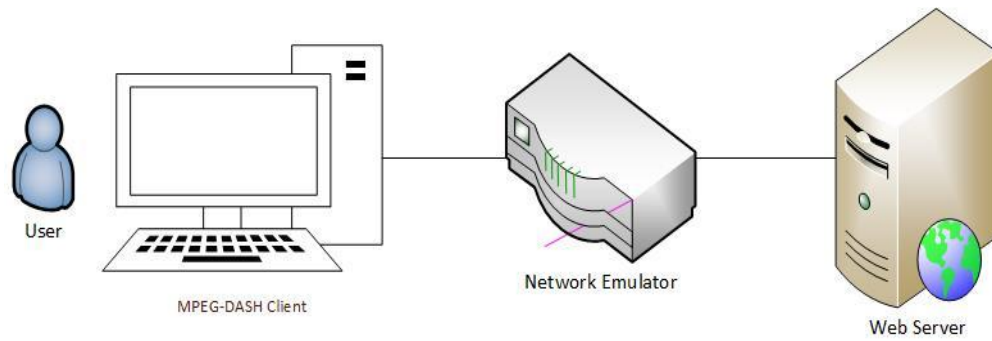


Figure 2. DASH Test bed

3.2 Experiments

The *BQMall* test sequence with a resolution of 832×480 was used for this experiment. The original YUV sequence was concatenated to create a longer video clip of 60 seconds in order to demonstrate the impact of switching between different bitrates. The H.264/AVC test sequences were encoded using the JM 18.0 reference software while the H.265/HEVC sequences were encoded using the HM10.0 reference software for Intra profile and Random Access (RA) profile, referred to as HEVC (Intra) and HEVC (RA) in the following, respectively. The test sequences were encoded at three different QP (Quantization Parameters) values of 28, 32, and 36 in order to get multiple representations with different bandwidths. The contents were segmented using the GPAC⁸ tool at 2s and 10s segment length respectively and then stored in the web server.

The streaming session starts by making an HTTP request to the server through the DASH client. Based on the request, an XML file is received showing the details of the media stored on the server where the client can now chooses its desired representation in light of its current network condition. Quality switching operations are subsequently triggered to adapt to the varying network conditions.

As mentioned, NETEM was utilized in the middle between the server and client (as illustrated in Figure 2) to emulate different WAN characteristics such as delay and packet loss. For our evaluation, we emulated six different network conditions as summarized in Table 1. We assumed the impairments were added at the start of the streaming session and remained throughout each streaming scenario. In order to measure the received picture quality, we compared the received video at the client with the reference YUV video sequence by measuring the PSNR for each test. The averaged PSNR results for each test are displayed graphically and analyzed in the next section.

Table 1. Streaming Scenarios.

ID	Description
1	No Packet loss or Delay
2	1% Packet loss
3	3% Packet loss
4	5% Packet loss
5	50ms Delay
6	100ms Delay
7	150ms Delay

3.3 Experimental Results

The experimental results for HEVC (RA), HEVC (Intra) and H.264/AVC under various network conditions are shown and analyzed in this section. Figure 3 and Figure 4 are results obtained from the experiments with varying packet loss ratios for 10s and 2s video segments, respectively; whilst Figure 5 and Figure 6 are for the experiments on various delays.

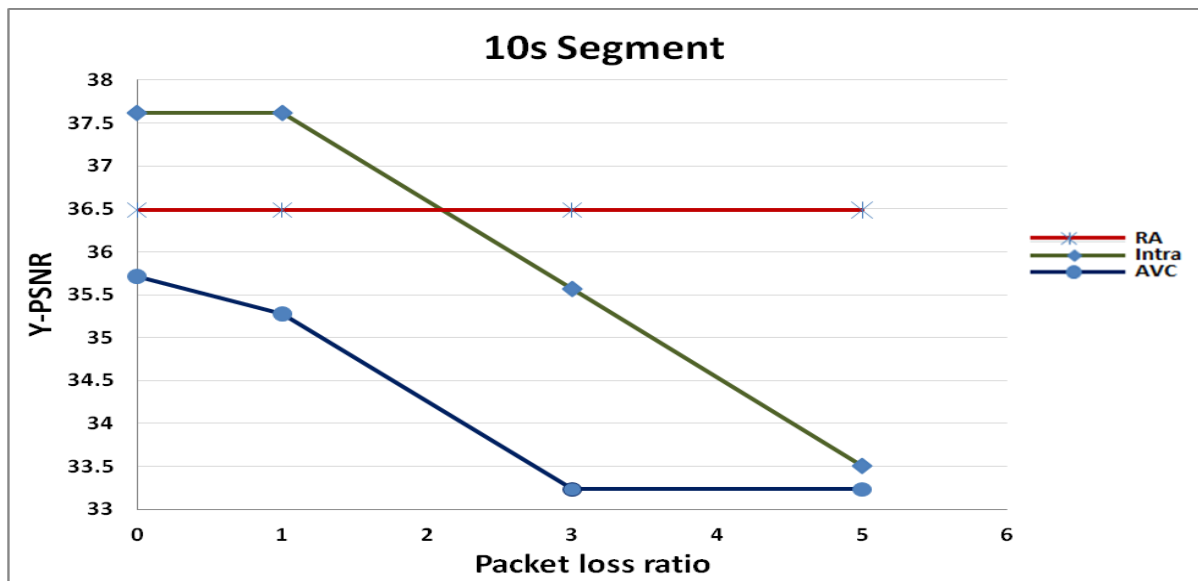


Figure 3. Y-PSNR comparison for different packet loss ratios (10s segment)

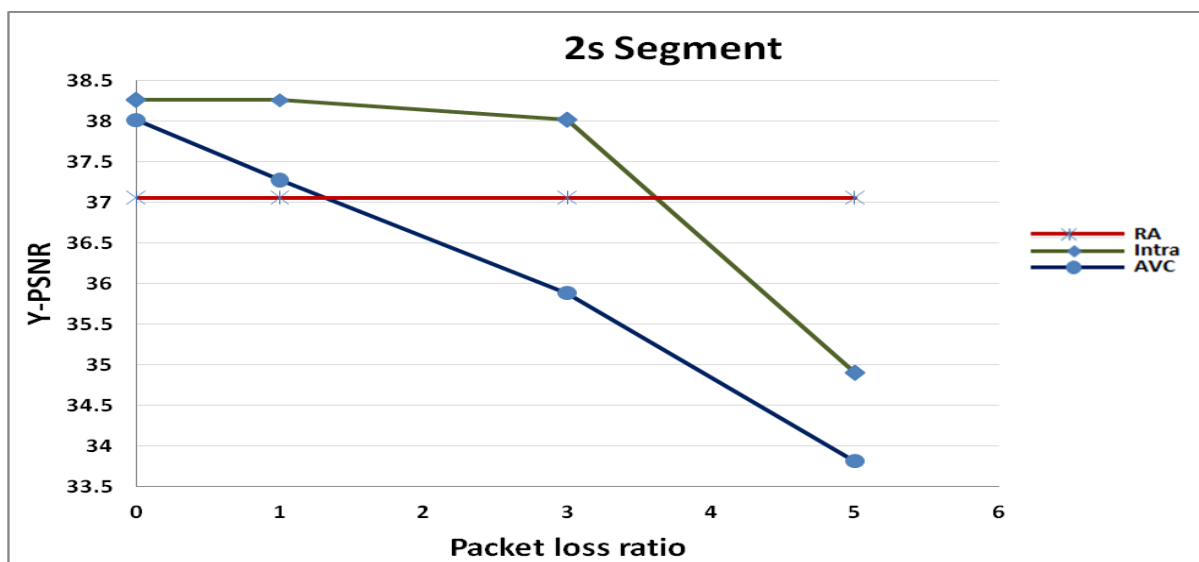


Figure 4. Y-PSNR comparison for different packet loss ratios (2s segment)

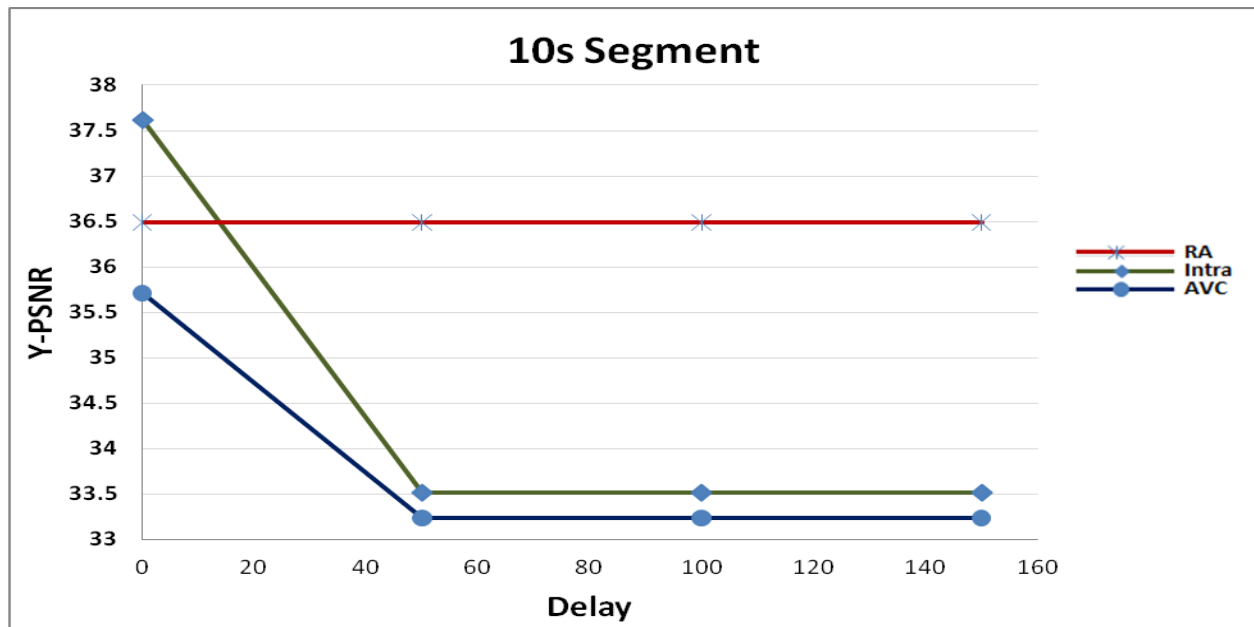


Figure 5. Y-PSNR comparison for different delays (10s segment)

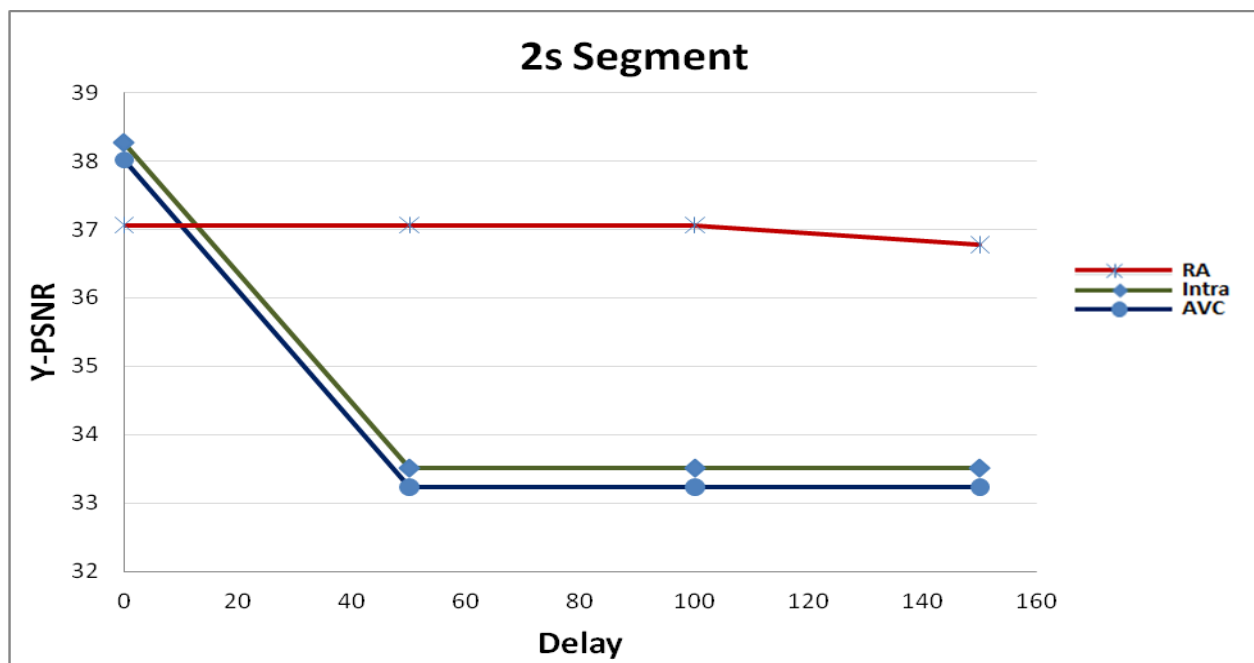


Figure 6. Y-PSNR comparison for different delays (2s segment)

Figure 3 shows the impact of packet loss on a 10s segment size DASH content for the H.264/AVC, H.265/HEVC Intra profile and Random Access profile, respectively. At 0% packet loss, the HEVC (Intra) system yielded the best PSNR of

37.61dB when compared with both HEVC (RA) and H.264/AVC showing improvements in PSNR of 1.13dB and 1.90dB respectively. With the introduction of non-zero packet loss, the PSNR values decreased for HEVC (Intra) and the H.264/AVC. However, the PSNR did not decrease for HEVC (RA). Although HEVC (RA) maintained the same PSNR, growing buffering delays were actually observed (due to the retransmission of lost packets) with the increase of the packet loss ratios, which would be reflected in complementary subjective video quality evaluations to be carried out in our future work. From Figure 3, it was also interesting to see that the PSNR of HEVC (Intra) dropped much more considerably by 2.05dB when the packet loss deteriorated (from 3% to 5%) in contrast to that of H.264/AVC. This can be perhaps attributed to the higher compression ratio achieved in HEVC (and thus fewer packets being transmitted, leading to more quality damage at the same packet loss level) compared with H.264/AVC.

From Figure 4, we can see the impact of a smaller segment size on the various codecs. At 0% packet loss, the PSNR of the H.264/AVC was increased by 2.3dB whilst the HEVC (Intra) and HEVC (RA) only increased by 0.65dB and 0.57dB respectively. Therefore, the smaller segment size resulted in a better performance compared with the larger segment size under various packet loss scenarios, and this effect was especially significant for H.264/AVC based DASH streaming. Furthermore, when Figure 3 and Figure 4 are compared at non-zero packet loss cases, higher picture quality was achieved with a smaller segment size for both HEVC (Intra) and H.264/AVC. This demonstrated the positive effect of more quality switching.

Results in Figure 5 and Figure 6 show the impact of delay on the two different segment sizes. The introduction of delay also had a negative effect on the streaming PSNR values of all codecs, although HEVC (both Intra and RA) produced higher PSNR values than H.264/AVC did. Both H.264/AVC and HEVC (Intra) reacted sharply to the 50ms delay by reduction in PSNR value of 2.47dB and 4.10dB respectively for the 10s segment, whilst the PSNR of the HEVC (RA) remained constant for the 10s segment but reduced a bit at the 150ms delay for the smaller segment size. From the results in Figure 5 and Figure 6, the delay had very similar impact on the PSNR of the two segment sizes for H.264/AVC and HEVC (Intra).

Overall, HEVC outperformed H.264/AVC in terms of PSNR. In particular, HEVC (Intra) demonstrated the best quality in comparison with H.264/AVC and HEVC (RA) without impairments. Under the conditions of various impairments, HEVC (Intra) was also better than H.264/AVC whilst HEVC (RA) was not affected by the impairments in terms of PSNR and only reacted at 150ms delay, although buffering was experienced when the network conditions worsened. In addition, smaller size segment always led to better performance for all codecs.

4. CONCLUSION

In this paper, we have focused on adaptive HTTP transmission of an HEVC video stream based on the MPEG DASH standard. We have established a practical test bed, and carried out an empirical evaluation of an HEVC-based DASH system using the PSNR objective quality metric. We have investigated the performance of the HEVC-DASH system under various network impairments in terms of packet loss and delay and compared its performance with an H.264/AVC based DASH system. Furthermore, we have compared the performance of these codecs with small and large video segment sizes, respectively.

Our experimental results have demonstrated that the HEVC (Intra) mode has a better quality when compared with H.264/AVC with or without network impairments. It has also been shown that the HEVC (RA) mode did not react to packet loss or delays under 150ms. Smaller segment size has yielded a better quality in comparison with the larger segment size, and this effect appears to be especially considerable for H.264/AVC. We have also observed the varying degree of buffering delays for the various codecs when the impairments were introduced and increased, which is an area worth more investigating. It will also be interesting to study the effects of different video content types on these codecs.

Our future work will further analyze the different behaviors of the DASH systems based on different codecs, investigate improved adaptation methods, and carry out further performance evaluation including subjective studies.

REFERENCES

- [1] Bossen, F., Bross, B., Suhring, K. and Flynn, D., "HEVC Complexity and Implementation Analysis," IEEE Transactions on Circuit and Systems for video technology 2(12), 1685-1696 (2012).

- [2] Sullivan, G.J., Ohm, J-R., Han, W-J. and Wiegand T., "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuit and Systems for video technology* 22(12), 1649-1668 (2012).
- [3] Pourazad, M.T. and Doutre, C., "HEVC: The Gold Standard for Video Compression," *IEEE Consumer Electronics Magazine* 1(3) 36-46 (2012).
- [4] Schierl, T., Hannuksela M.M., Wang, Y-K. and Wenger S., "System Layer Integration of High Efficiency Video Coding," *IEEE Transactions on Circuit and Systems for video technology* 22(12), 1871-1884 (2012).
- [5] Zambelli, A., "IIS Smooth Streaming Technical Overview," Microsoft Corporation.
<http://www.microsoft.com/en-us/download/confirmation.aspx?id=17678> (last access: March 2014).
- [6] Muller, C., Lederer, S. and Timmerer, C., "An Evaluation of Dynamic Adaptive Streaming over HTTP in Vehicular Environments," *Proc. ACM Multimedia System Conference and 4th ACM Workshop on Mobile Video*, (2012).
- [7] Stockhammer, T., "Dynamic Adaptive Streaming over HTTP – Standards and Design Principles," *Proc. ACM Conference on Multimedia Systems*, 133-144 (2011).
- [8] GPAC. <<http://gpac.wp.mines-telecom.fr/>>
- [9] Apple HTTP Live Streaming. <<https://developer.apple.com/streaming/>>
- [10] Lederer, S., Muller, C. and Timmerer, C., "Dynamic Adaptive Streaming over HTTP Dataset," *Proc. ACM Conference on Multimedia Systems*, 89-94 (2012).
- [11] Ognenoski, O., Razaak, M., Martini, M.G. and Amon, P., "Medical Video Streaming Utilizing MPEG-DASH," *IEEE 1st International Workshop on Active and Health Ageing*, 54-59 (2013).
- [12] Ognenoski, O., Martini, M.G. and Amon, P., "Segment-based Teletraffic Model for MPEG-DASH," *IEEE International Workshop for Multimedia Signal Processing*, 333-337 (2013).
- [13] Mok, R.K.P., Chan, E.W.W. and Chang, R.K.C., "Measuring the Quality of Experience of HTTP Video Streaming," *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 485-492 (2011).
- [14] Thang, T.C., Ho, Q-D., Kang J-W., and Pham, A.T., "Adaptive Streaming of Audio Visual Content using MPEG-DASH," *IEEE Transactions on Consumer Electronics* 58(1) 78-85 (2012).
- [15] Martin, J., Fu, Y., and Wourms, N., "Characterizing Netflix Bandwidth Consumption," *IEEE Consumer Communications and Networking Conference*, 230-235 (2013).
- [16] Gouache, S., Bichot, G., Bsila, A. and Howson, C., "Distributed & Adaptive HTTP Streaming," *IEEE International Conference on Multimedia and Expo (ICME)*, 1-6 (2011).
- [17] He, Y., Kunstner, M., Gudumasu, S., Ryu, E-S., Ye, Y. and Xiu, X., "Power Aware HEVC Streaming for Mobile," *IEEE Conference on Visual Communications and Image Processing (VCIP)*, 1-5 (2013).
- [18] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2013-2018.
<http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.html>.
- [19] Nightingale, J., Wang, Q. and Grecos, C., "HEVStream: a framework for streaming and evaluation of high efficiency video coding (HEVC) content in loss-prone networks," *IEEE Trans. Consum. Electron.*, 58(2), 404-412 (2012).
- [20] Basak, O., Pourazad, M.T., Nasiopoulos, P. and Leung, V.C.M., "A study on the HEVC performance over lossy networks," *Proc. 19th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, 785-788 (2012).
- [21] Nightingale, J., Wang, Q. and Grecos C., "Priority-based methods for reducing the impact of packet loss on HEVC encoded video streams," *Proc. SPIE Real-Time Image and Video Processing 2013* (2013).
- [22] Sanchez, Y., Schierl, T., Hellge, C., and et al., "Efficient HTTP-based Streaming using Scalable Video Coding," *Elsevier Signal Processing: Image Communication*, 27(4), 329-342 (2011).