Visión Electrónica Vol. 13 No. 1 (2019) • January-June • p.p. 152-161 • ISSN 1909-9746 • ISSN-E 2248-4728 • Bogotá (Colombia)

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Visión Electrónica
Más que un estado sólido
https://revistas.udistrital.edu.co/index.php/visele/index

Visión Electrónica

A CASE-STUDY VISION

Adaptive video transmission over software defined networks

Trasmisión adaptativa de video sobre redes definidas por software

Fran Wilson Sanabria Navarro [1]; Juan Gabriel Bustos [2]; Wilder Eduardo Castellanos Hernández [3]

INFORMACIÓN DEL ARTÍCULO

Historia del artículo:
Enviado: 02/12/2018
Recibido: 11/12/2018
Aceptado: 20/12/2018

Keywords:
Adaptive Video Streaming
DASH
Multimedia Communication
Software Defined Networks

ABSTRACT

This paper presents the results of a study on the evaluation of adaptive transmission of video streams using the DASH technique on Software Defined Networks. There are also presented in this document, the description of the tools required for the implementation of the evaluation, as well as a description of the methodology used for the development of the experiments. In addition, the results of an adaptive transmission of a video by using DASH are presented. This transmission was carried out over a software defined network emulated on MININET. The results show that DASH technique easily allows to implement video streaming services that can adapt the quality of the transmission according to the resources available in the network.

Palabras clave:
Video Adaptativo
DASH
Comunicaciones multimedia
Redes definidas por software

RESUMEN

En este artículo se presentan los resultados de un estudio relacionado con la evaluación de la transmisión adaptativa de flujos de video usando el estándar DASH sobre escenarios de redes definidas por software. Dentro de los aspectos que se presentan en este documento está la descripción de las herramientas software necesarias para la implementación de la evaluación, así como la metodología de uso de estas. Además, se presentan los resultados de un experimento de emulación de una topología de red definida por software en la plataforma MININET y la transmisión adaptativa de un video mediante DASH. Los resultados muestran que la técnica DASH permite fácilmente la implementación de servicios de video streaming que son capaces de adaptarse a los recursos disponibles en la red. También se resalta la facilidad de experimentar con las redes definidas por software en la plataforma de emulación utilizada y la configuración de servicios multimedia sobre este tipo de redes.

[1] BSc. In Electronic engineering, Universidad de San Buenaventura, Colombia. Research assistant, Universidad de San Buenaventura, Colombia. E-mail: sfran@academia.usbbog.edu.co ORCID: http://orcid.org/0000-0002-1844-7789

[2] BSc. In Electronic engineering, Universidad de San Buenaventura, Colombia. Research assistant, Universidad de San Buenaventura, Colombia. E-mail: jbustos@academia.usbbog.edu.co ORCID: https://orcid.org/0000-0001-7458-6321

[3] Ph.D. In Telecommunications, MSc. In Technologies, Systems and Communications Networks, Universitat Politècnica de València, España. BSc. In Electronic engineering, Universidad Industrial de Santander, Colombia. Current position: Professor at Universidad de San Buenaventura, Colombia. E-mail: wcastellanos@usbbog.edu.co ORCID: https://orcid.org/0000-0003-0311-492X

Cite this article as: F. W. Sanabria-Navarro, J. G. Bustos; W. E. Castellanos-Hernández, "Adaptive video transmission over software defined networks", Visión electrónica, algo más que un estado sólido, vol. 13, no. 1, january-june 2019, pp. 152-161, DOI: https://doi.org/ 10.14483/22484728.14398

1. Introduction

The massive demand for video streaming services such as Netflix, YouTube, iTunes and BitTorrent Service, has led to the increase in the video traffic carried by the networks. This fact has motivated researchers and network operators to development solutions in order to improve the transport of these flows through data networks. Several aspects are relevant in this matter, mainly those that are related to offer the video services with the quality that users demand. Even though their connections do not have an efficient management of the quality of communications. In these scenarios, the ability to adapt the video streams according to the network conditions is essential. In this sense several solutions have been proposed in recent years, such as scalable coding and adaptive transmission techniques.

On the other hand, with the inevitable implementation of SDNs (Software Defined Networks) by network operators is necessary to start with the evaluation of the adaptive video techniques in this new network model. The change of paradigm, proposed by SDNs, involves the transfer of the control plane, currently located in physical devices (routers and switches), unifying it in a single element external to the physical network, called the controller [1]. This fact implies the implementation of a new software component in charge of the tasks of network routing and controlling, instead of traditional distributed protocols such as BGP (Border Gateway Protocol) or OSPF (Open Shortest Path First). The main characteristic of this new software, called Controller, is that involves open standard management software which is non-dependent of the manufacturer of the network devices.

Within this paper describes the results of a study in which the adaptive transmission of videos was evaluated using the DASH (Dynamic Adaptive Streaming over Hypertext Transfer Protocol) technique. The type of networks selected to carry out the transmission of the videos was a Software Defined Network (SDN). The study was developed over the emulation platform MININET [2], on which a network topology was defined and video content was transmitted. We set different networks scenarios where the transmission capacity was varied in order to evaluate the ability of DASH to adapt the video flow to the available bandwidth. After video transmission, the Peak Signal Noise Rate (PSNR) was calculated to quantify the quality of the received videos.

The rest of the paper is organized as follows. First, it is introduced a brief study of the state of the art on adaptive video transmission in Section 2. Then, in Section 3 it is given a brief introduction to the DASH technique and the Software Defined Networks. The simulation scenarios are described in Section 4 and in Section 5 are shown the results of the performance evaluation. Finally, are presented the conclusions in Section 6.

2. Related Works

Recently there has been a notable interest on the development of studies and applications related to the transmission of digital video on different types of networks [3–5]. One of the problems that has most focused the attention of researchers is how to adapt the video streams according to the variable conditions of the networks. Several adaptive techniques have been developed, which need to estimate the available resources (for example, in terms of bandwidth or CPU capacity) in order to adapt the video quality that is provided to the user. Mainly, there are different types of techniques for adaptive streaming [6,7]. One alternative is the transcoding method, which consists on transcode from one encoding to another to match with a specific bit rate. This technique is highly costly due to the fact that the raw video content requires to be adapted several times for different user requests [8]. The second alternative for adaptive transmission employs Scalable Video Coding [9,10] which has attracted great interest by researchers to improve video streaming over scenarios where the resources are very limited, as is studied in [11,12]. The third technique for adaptive video transmission is based on the idea of stream switching. This technique encodes the raw video at several different increasing bit rates, generating n versions of the same video. In addition, an algorithm must regularly choose the video version that matches the available bandwidth of the communication link. An example of this type of technique is MPEG-DASH (Dynamic Adaptive Streaming over HTTP), which has become a standard that aims to provide video streaming over heterogeneous devices and fluctuant network conditions. References [13–16] evaluate some advantages of using this method for adapting video streams to network variations. In the same way, integration of scalable techniques and DASH has been considered as a more robust and flexible solution, as described in [3,17,18]. Also, the transmission of video on network scenarios easier to configure and emulate has become an important aspect nowadays. For example, the study described in [19] offers a vision on the transmission of adaptive video on Software Defined Networks. There are few works focused on evaluating the aspects related to the transmission of video in the SDNs [14, 20]. These works only studied traditional techniques for video streaming, while the more advanced techniques still need more evaluation.

3. Background

3.1. Digital Video Transmission

As first instance, before transmitting a video sequence through a data network, it is necessary to establish an adequate format to decrease the file size and bit rate. When performing the compression of a video, it generally exploits two types of redundancies: firstly, there is spatial redundancy which correlates the similarity between neighboring pixels in a frame and that is reduced by means of intra-frame coding. Secondly, there is temporal redundancy which relates the similarities between consecutive frames [21]. Figure 1 briefly describes the main functional blocks of a video encoder. The two main objectives of video encoders are to achieve compression efficiency (representing the video with the least number of bits) and to encode frames with a high quality.

Different standards for video coding have been developed, currently the most widely used is the H.264/MPEG-4 standard, which is applied in different scenarios such as digital television, interactive multimedia and audiovisual communications on different network technologies. Recently, the H.265/ HEVC (High Efficiency Video Coding) standard has been developed with the purpose of facing the challenge of ultra high definition content distribution. Unlike the H.264 standard, where each frame is divided into macroblocks of 16x16 pixels, HEVC replaces the macroblocks with tree coding units (CTU, Coding Tree Unit) which allows encoder to use blocks with larger structures. The maximum size of these CTUs can be up to 64x64 pixels [22]. The use of large encoding units on the homogeneous regions of a video that has little or no movement between two adjacent frames leads to a significant compression gain.

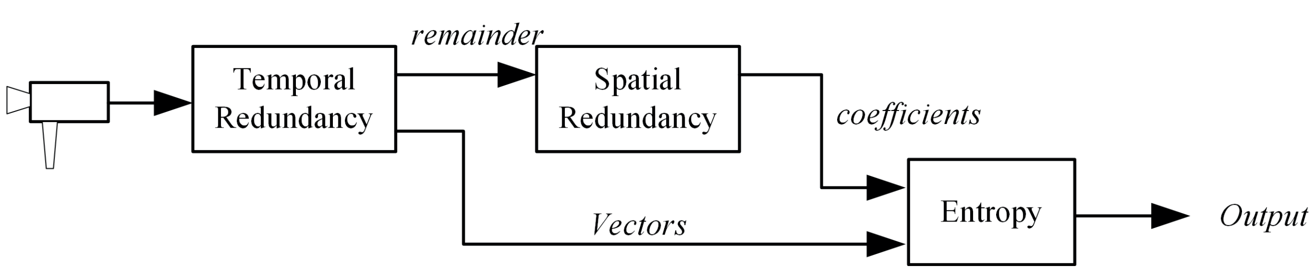
The MPEG-DASH standard is also used to perform video streaming, this allow HTTP servers to adapt the video stream to the network conditions of the users, as will be described later.

3.2. Software Defined Networks, SDN

The concept of SDNs involves a new vision about the network management and its main characteristic is the transfer of the management tasks from the control layer of physical devices to a new element called controller. This controller allows network administrator to configure network devices such as switches, routers or firewalls from only one network point.

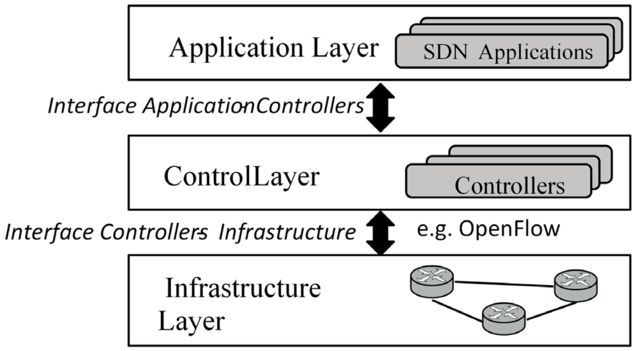
In this way, network manager can have a complete view of the network because information is constantly exchanged between the different layers of the model, which are: application layer, control layer, and infrastructure layer, as can be seen in the Figure 2 [23]. Recently, several protocols have been standardized for SDNs, for example the OpenFlow protocol, which establish the connection between control layer and infrastructure layer of the network devices.

Figure 1. Block diagram of a video encoder.



Source: own.

Figure 2. Reference Model for SDNs [24].



One of the most platforms used for experimentation with SDNs is MININET, which is a network emulator that allows users to configure a set of switching nodes, links and controllers in a single Linux kernel. The main advantage of using MININET is the speed and ease of implementing experiments with SDNs and OpenFlow. Thus, MININET has become an essential tool in which new ideas can be developed and tested before being implemented in a real or physical environment.

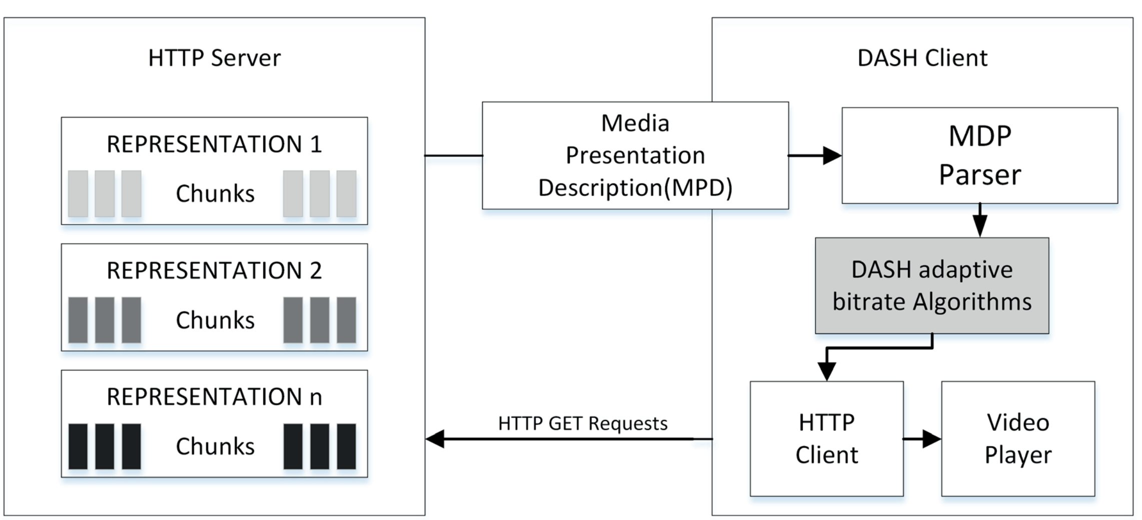
3.3. MPEG-DASH (Dynamic Adaptive Streaming over HTTP)

The MPEG-DASH coding standard enable the transmission of adaptive video over HTTP. It was developed by the Moving Picture Expert Group (MPEG) [25] in order to solve the problem to deliver the content adapted to the multiple platforms and consumer electronic devices [26]. The DASH standard defines that the control over the video stream is exclusively performed by the client [25]. This involves the use of a DASH client, which is also responsible for the reproduction of the video. This client is in charge of requesting from the server the information required to carry out the video streaming. On account of this, video content must be encoded using the DASH standard. When a video is encoded in DASH a file called *manifest* (or *mdp*) is generated and stored in the HTTP server. In addition, different representations of the video are also stored. These representations are different versions of the original video, but with different qualities. At the same time, each video representation is subdivided into segments of a short duration. In this way, if network resources are very restrictive, a client can request from the server the segments of the representation of the lowest quality. On the contrary, when the network has more available resources, a client can request the segments of the representation of best quality. In Figure 3 architecture of a video transmission scenario using DASH is described.

4. Evaluation of the adaptive video transmission over SDN

In this section we describe how the evaluation of video transmission using DASH over software defined networks was carried out. The methodology used in our experiments is shown in Figure 4 and it is described below. The first step is to encode the original video sequence according to H.264/AVC standard using the FFMPEG software [28]. The video sequence used as traffic flow was the well-known test sequence known as STUDENTS, which belongs to the gallery of test sequences that can be

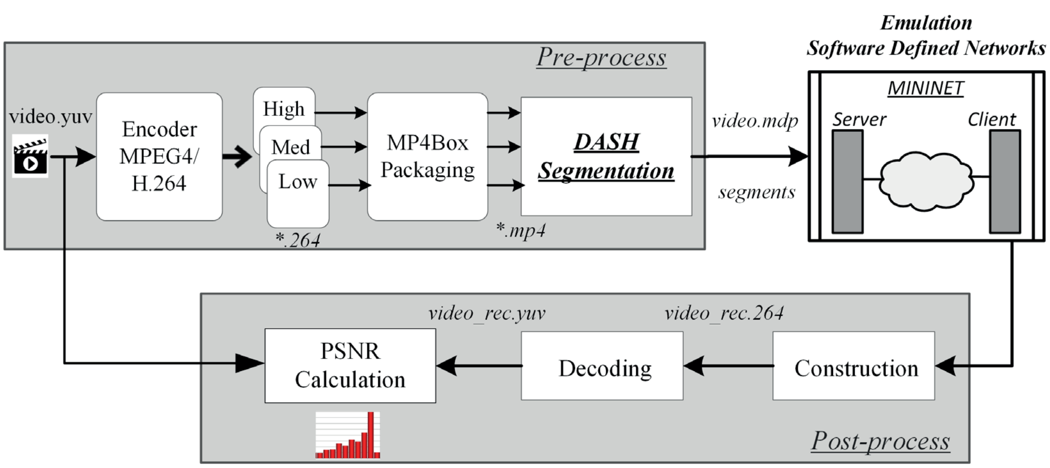
Figure 3. DASH client-server architecture [27].



consulted in [29]. The original video in YUV format (RAW, without compression) is composed of 1007 frames and has a frame size of 352x288 pixels. This video was encoded with three different levels of compression (CRF, Constant Rate Factor) in order to obtain three versions or representations of the original video. Therefore, each representation of the video has different quality: the video encoded with *crf = 40* corresponds to a low quality, *crf = 20* to medium quality and *crf = 10* to a high quality. Subsequently, each version of the video was packaged in an MP4 container using the MP4Box software [30]. After packaging, the DASH segments and the *"manifest"* file (.mpd) are generated. Because each representation of the video has a duration of 40 seconds, DASH segments of 10 seconds of duration were generated, obtaining four segments for each representation. Such as is graphically illustrated in Figure 5, in the "*manifest*" file the bit rates necessary to display each representation of the videos are described. For example, the video representation with the worst quality require 25,2 Kbps, 309 Kbps for the medium quality representation and 1,855 Mbps for the best quality. After video segments and *.mdp* file are generated, the video can be transmitted through the SDN.

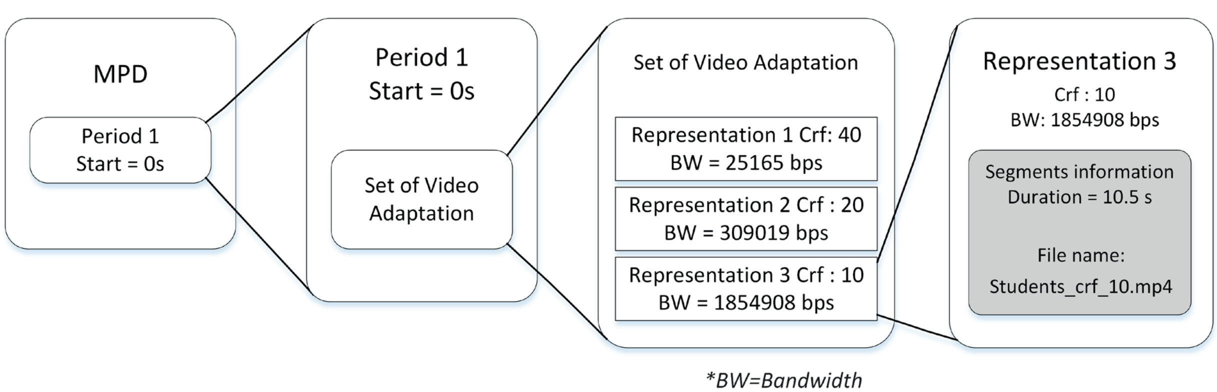
Once video transmission through the network has finished, the received video is formed from the segments transmitted by the HTTP server. Later, the conformed video is decoded to YUV format to compare it with the original RAW video in order to measure the PSNR (Peak Signal Noise Rate) for evaluating the end-to-end delivered video quality. PSNR is an objective metric used to estimate the image quality of a video in comparison with its original version. Therefore, the

Figure 4. Methodology used in experiments.



Source: own.

Figure 5. Description of the "manifiest" file.



Source: own.

calculation of the PSNR gives us a clear quantification of the distortion suffered by the video. This metric is one of the most used to evaluate the quality of a video after its transmission over a network. Equation (1) shows the definition of the PSNR.

$$PSNR = 10 \cdot Log_{10}\left(\frac{255^2}{MSE}\right) \quad donde,$$
$$MSE = \frac{1}{NM}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|I_{org}(m,n) - I_{rec}(m,n)\right|^2 \quad (1)$$

Where, *Iorg* is the original frame and *Irec* is the frame received after being transmitted by a network; *M*, *N* is the size of the frame and *MSE* is the mean square error. According to (1), the PSNR is the comparison, pixel by pixel, of each frame contained in the two videos (original and received). In this way, a numerical value is obtained which reflects the distortion suffered by the received video regarding the original.

4.1. Simulation Scenario

The network topology emulated in MININET is shown in Figure 6. The network consists on 6 hosts (*h1, h2, h3, h4, h5, h6*) and 3 switches (*S1, S2* and *S3*). All links were configured with a transmission capacity of 2 Mbps. The source node (server) is h2 and the client node is h5. This network was programmed as a Python script and a fragment of the script is shown in the right part of Figure 6.

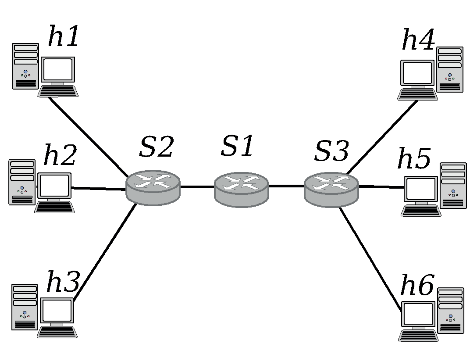
The DASH-video was transmitted over three emulated scenarios. All scenarios involve the same network topology but with different link capacity between client node (h5) and switch 3 (S3). Thus, a bottleneck is generated in order to force an adaptive process during video transmission. In the first scenario the link was configured to support a bandwidth of only 200Kbps, in the second to 400Kbps and in the third scenario to 2Mbps. With the different values of capacity in the connection link of the client, we aimed to evaluate the adaptability of the DASH server. That is, the server should transmit the suitable video segments according to the available bandwidth in the link.

5. Results

After transmitting the video sequence in the first scenario (bandwidth set to 200Kbps), received video was made up of the 4 segments with the worst quality, as can be seen in Figure 7. This is due to the fact that bit rate required by other video representations is higher than the bandwidth of the link.

In the second network scenario (i.e. where the client link was configurated to 400Kbps), received video is composed of two segments of poor quality and the last two of medium quality. This was expected since with the available bandwidth of this scenario only medium quality segments can be delivered because they required a link capacity of 309Kbps, as was described in Section 4. It is important to note that although the network has enough bandwidth to initiate the video transmission delivering medium quality segments from the beginning. However, default settings of DASH define that transmission should start with the delivery of the lower-quality segments.

Figure 6. Network topology and script.

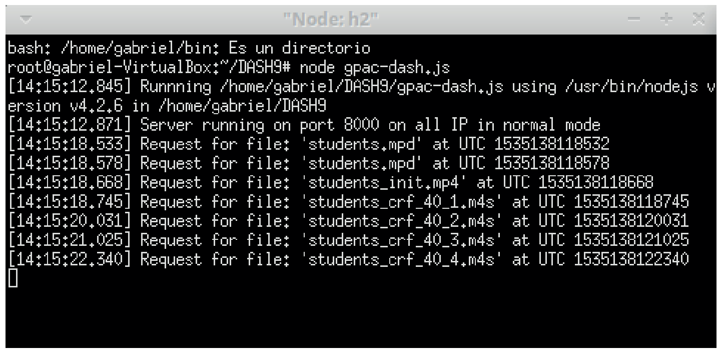



```
print "*** Creating links"

net.addLink(s2,s1,bw=2,delay='1ms',loss=0)
net.addLink(s1,s3,bw=2,delay='1ms',loss=0)
net.addLink(h1,s2,bw=2,delay='1ms',loss=0)
net.addLink(h2,s2,bw=2,delay='1ms',loss=0)
net.addLink(h3,s2,bw=2,delay='1ms',loss=0)
net.addLink(s3,h4,bw=2,delay='1ms',loss=0)
net.addLink(s3,h5,bw=2,delay='1ms',loss=0)
net.addLink(s3,h6,bw=2,delay='1ms',loss=0)

print "*** Starting network"

net.build()
```

Source: own.

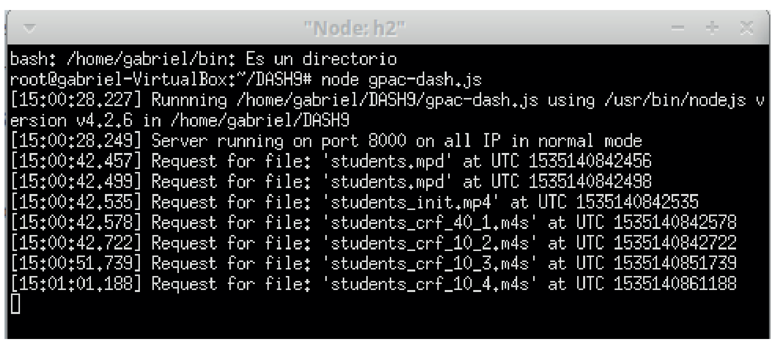
Finally, video transmission was performed in the third scenario (available bandwidth equal to 2Mbps). In this case, server delivered the first segment of poor quality (crf=40) and next three segments of high quality (crf=10), as evidenced in Figure 8. As an example of the video transmission process, in Figure 10 we described the exchange of messages between client and server during the execution of scenario 3.

Figure 7. Video segments requeted by client player in the scenario with a link of 200 Kbps.



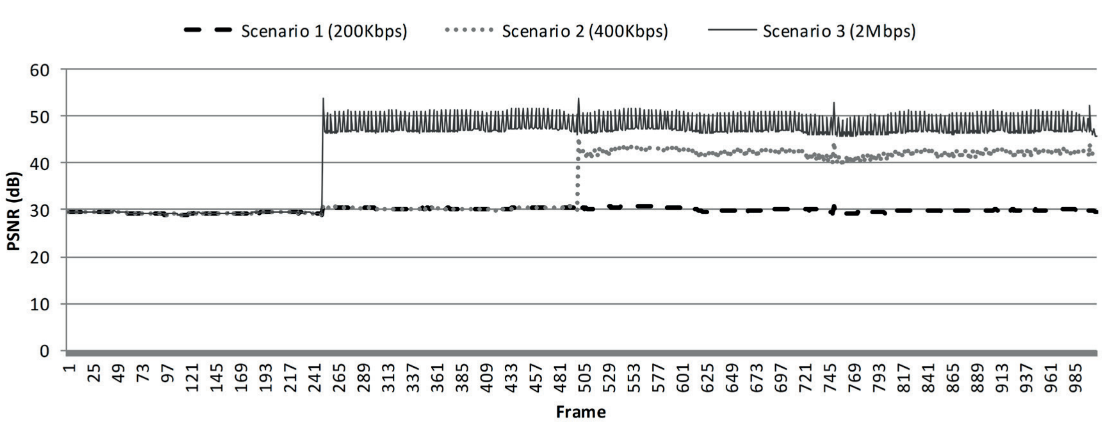
Source: own.

Figure 8. Video segments requeted by client player in the scenario 3 with a link of 2 Mbps.



Source: own.

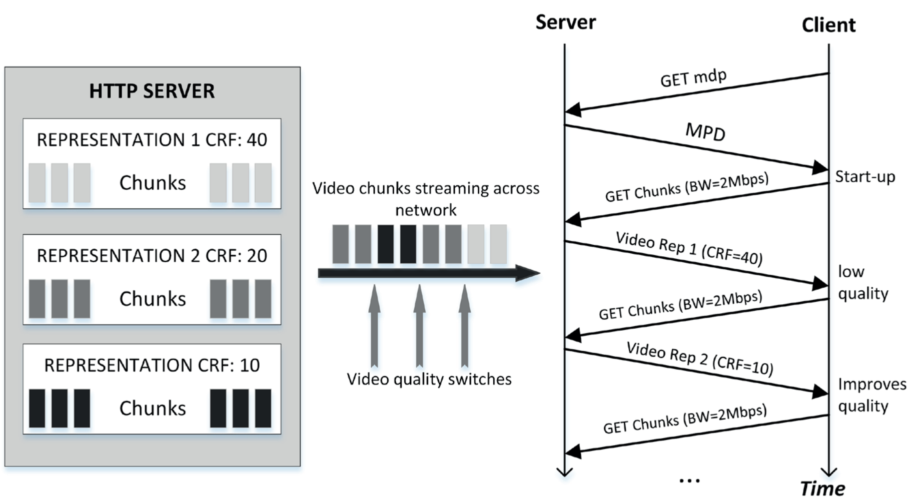
Figure 9. PSNR of received videos.



Source: own.

After video transmission over the three network scenarios, the PSNR was calculated. PSNR values allow us to estimate changes in the video quality perceived by users. Figure 9 shows the PSNR obtained per frame for each video stream obtained in the three scenarios. The average PSNR obtained in the first scenario was 29,92 dB, 35,98 dB in the second scenario and 43,24 dB in the third scenario. These values are consistent with the segments transmitted in each scenario. As shown in Figure 10, in the first scenario are transmitted the lower quality segments since the bit rate required to stream segments of medium or high quality is greater than the capacity of the link. On the other hand, in scenarios two and three (400Kbps and 2Mbps, respectively) it is observed that HTTP server delivered segments with the maximum quality that the link can support. For example, in the second scenario after streaming the first two segments of low quality (corresponding to 504 frames) the quality was increased about 12 dB (segments of medium quality). Something similar happens in the third scenario, after receiving the first segment of low quality server transmits the DAHS segments of high quality increasing the PSNR from about 30dB to 45dB. These results are consistent with the operation of DASH as well as its ability to adapt the video flows according to network state was also verified.

Figure 10. Graphic description of video streaming using DASH standard (scenario 3) [27].



6. Conclusions

In this article, was propose a methodology in order to guide the implementation of studies that involve adaptive video transmission using the DASH standard on Software Defined Networks. In addition, we present the results obtained after transmitting a video sequence according to MPEG-DASH standard over three network scenarios. The differences between scenarios was the value of the available bandwidth on the client connection. Through this simple experiment it was possible to verify the operation of DASH on a SDN as well as the interaction of the different components of an adaptive video streaming service.

In terms of video quality, the obtained PSNR values were coherent with the DASH segments that were transmitted from HTTP server. Based on these results, the correct functioning of the mechanisms used by DASH standard was demonstrated. However, in scenarios where the capacity of links or nodes changes rapidly, the adaptation process of the video flow can be slow because the speed of adaptation in DASH depends on the duration of the segments.

On other hand, the experimentation with SDNs and MININET, allowed to identify advantages that SDNs offer to the provision of video services. For instance, centralizing control tasks of the entire network in a single protocol and a single layer, facilitates the implementation of the resource estimation algorithms and mechanisms for adapting the multimedia flows to the capacity of the network. This allowed us to identify new alternatives that will be used for the development of more advanced studies in which scalable coding and DASH can be combined.

References

[1] Open Networking Foundation, "Software-Defined Networking", Palo Alto, CA, 2012.

[2] B. Lantz, N. Handigol, B. Heller and V. Jeyakumar, "Introduction to Mininet", 2017. [Online]. Available: https://github.com/mininet/mininet/wiki/Introduction-to-Mininet

[3] C. Müller, D. Renzi, S. Lederer, S. Battista and C. Tinnerer, "Using Scalable Video Coding for Dynamic Adaptive Streaming over HTTP in mobile environments", Proceedings of the 20th European Signal Processing Conference (EUSIPCO), 2012, pp. 2208–2212.

[4] S. Memon, et al., "Performance Evaluation of Video Streaming Over Wired Networks Using NS-2 Simulator", First International Conference on Modern Communication & Computing Technologies (MCCT'14), Nawabshah, Pakistan, 2014.

[5] D. Migliorini, E. Mingozzi and C. Vallati, "QoE-Oriented Performance Evaluation of Video Streaming over WiMAX", Proceedings of the 8th international conference on Wired/Wireless Internet Communications, 2010, pp. 240–251, https://doi.org/10.1007/978-3-642-13315-2_20

[6] K. Evensen et al., "Adaptive Media Streaming to Mobile Devices: Challenges, Enhancements, and Recommendations", *Advances in Multimedia*, vol. 2014, 2014. https://doi.org/10.1155/2014/805852

[7] C. Timmerer, et al., "Guest Editorial Adaptive Media Streaming", *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 4, 2014, pp. 681–683, https://doi.org/10.1109/JSAC.2014.140401

[8] C. Zhu and Y. Li, "Advanced Video Communications over Wireless Networks", Boca Raton: CRC Press, 2013, https://doi.org/10.1201/b13746

[9] C. Singhal and A. K. Sahu, "UHD video transmission using adaptive SHVC in wireless networks", 20th International Symposium on Wireless Personal Multimedia Communications (WPMC), 2017, pp. 234–240. https://doi.org/10.1109/WPMC.2017.8301815

[10] W. Castellanos, J. C. Guerri and P. Arce, "Performance evaluation of scalable video streaming in mobile ad hoc networks", *IEEE Latin America Transactions*, vol. 14, no. 1, 2016, pp. 122–129. https://doi.org/10.1109/TLA.2016.7430071

[11] W. Castellanos, "Quality of service routing and mechanisms for improving video streaming over mobile wireless ad hoc networks", thesis Ph.D., Universitat Politècnica de València, Valencia, España, 2015.

[12] B. Oztas, M. Pourazad, P. Nasiopoulos and V. Leung, "A study on the HEVC performance over lossy networks", 19th IEEE International Conference on Electronics, Circuits and Systems (ICECS), 2012, pp. 785–788. https://doi.org/10.1109/ICECS.2012.6463542

[13] A. Zabrovskiy, E. Petrov, E. Kuzmin and C. Timmerer, "Evaluation of the Performance of Adaptive HTTP Streaming Systems", 2017. [Online]. Available: https://arxiv.org/abs/1710.02459v1

[14] J. Quinlan, A. Reviakin, A. Khalid, K. Ramakrishnan and C. Sreenan, "D-LiTE-ful: An evaluation platform for DASH QoE for SDN-enabled ISP offloading in LTE", Proceedings of the Tenth ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation, and Characterization, 2016, pp. 91–92. https://doi.org/10.1145/2980159.2980175

[15] R. Stobiecki, "The control system for the service of adaptive video streaming services offered in the cloud computing environment", thesis, The Institute of Telecommunications (FEIT / IT), 2017.

[16] T. C. Thang, Q. Ho, J. won Kang and A. Pham, "Adaptive streaming of audiovisual content using MPEG DASH", *IEEE Transactions on Consumer Electronics*, vol. 58, no. 1, 2012, pp. 78–85. https://doi.org/10.1109/TCE.2012.6170058

[17] S. H. Yang and H. B. Chen, "Hybrid unicast and multicast video streaming using H. 264 SVC and FLUTE for set-top box users", 7th International Symposium on Next Generation Electronics (ISNE), 2018, pp. 1–4. https://doi.org/10.1109/ISNE.2018.8394736

[18] S. Mori and M. Bandai, "QoE-aware quality selection method for adaptive video streaming with scalable video coding", IEEE International Conference on Consumer Electronics (ICCE), 2018, pp. 1–4. https://doi.org/10.1109/ICCE.2018.8326093

[19] A. Zabrovskiy, E. Kuzmin, E. Petrov and M. Fomichev, "Emulation of dynamic adaptive streaming over HTTP with Mininet", 18th Conference of Open Innovations Association and Seminar on Information Security and Protection of Information Technology (FRUCT-ISPIT), 2016. https://doi.org/10.1109/FRUCT-ISPIT.2016.7561555

[20] D. Stohr, et al., "QoE Analysis of DASH Cross-Layer Dependencies by Extensive Network Emulation", Proceedings of the workshop on QoE-based Analysis and Management of Data Communication Networks, 2016, pp. 25-30, https://doi.org/10.1145/2940136.2940141

[21] B. Bing, "Next-Generation Video Coding and Streaming. Hoboken", United Stated: John Wiley & Sons, Incorporated, 2015, https://doi.org/10.1002/9781119133346

[22] G. J. Sullivan, J. Ohm, W. Han and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard", *IEEE Transactions on circuits and systems for video technology*, vol. 22, no. 12, 2012, pp. 1649–1668. https://doi.org/10.1109/TCSVT.2012.2221191

[23] D. Serrano, "Redes Definidas por Software (SDN): OpenFlow", thesis, Universitat Politècnica de València, Valencia, 2015.

[24] W. Xia, Y. Wen, C. Foh, D. Niyato and H. Xie, "A Survey on Software-Defined Networking", *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, 2015, pp. 27–51. https://doi.org/10.1109/COMST.2014.2330903

[25] B. Casin Nuñez, "DASH: Un estándar MPEG para streaming sobre HTTP", thesis, Universidad Politécnica de Cataluña, Barcelona, 2013.

[26] A. Pennington, "MPEG-DASH: Making Tracks Toward Widespread Adoption", *Streaming Media Magazine*, no. 3985, 2013.

[27] J. Kua, G. Armitage and P. Branch, "A Survey of Rate Adaptation Techniques for Dynamic Adaptive Streaming Over HTTP", *IEEE Communications Surveys Tutorials*, vol. 19, no. 3, 2017, pp. 1842–1866. https://doi.org/10.1109/COMST.2017.2685630

[28] F. Bellard, "FFmpeg", 2018. [Online]. Available: https://www.ffmpeg.org/

[29] Xiph.Org Foundation, "Video Test Media. Derf Collection". [Online]. Available: https://media.xiph.org/video/derf/

[30] J. Le Feuvre, "MP4Box | GPAC". [Online]. Available: https://gpac.wp.imt.fr/mp4box/