

Document downloaded from:

<http://hdl.handle.net/10251/101674>

This paper must be cited as:



The final publication is available at

<https://doi.org/10.1002/dac.3551>

Copyright John Wiley & Sons

Additional Information

Adaptive Video Streaming Testbed Design for Performance Study and Assessment of QoE

Miran Taha^{1,2}, Jaime Lloret¹, Aree Ali^{2,3}, Alejandro Canovas¹

¹Universitat Politècnica de València, Spain

²University of Sulaimani, Kurdistan region, Iraq

³University of Human Development, Kurdistan region, Iraq

miab2@doctor.upv.es, jlloret@dcom.upv.es, aree.ali@univsul.edu.iq, alcalosol@posgrado.upv.es

Abstract— Video streaming applications have advanced significantly with the development of the Internet. HTTP Adaptive Streaming (HAS) as a streaming technology allows adaptation of downloaded video quality to current network conditions. Thus, the adaptation technique of HAS application considers switching between different qualities of the video and avoids stalling streamed frames over high oscillation client's throughput. This is provided satisfactory of the users' experience to perceive quality as the known quality of experience (QoE). Therefore, there are high competitions of service providers to attract costumers to use services and QoE is become a more important parameter to lead the service providers to know about the end-users feedback. Implementing HAS applications to find out QoE in real life scenarios of vast networks becomes more challenging and complex task regards to cost, agile, time and decisions. In this paper, a virtualized network testbed to virtualize various machines to support implementing experiments of adaptive video streaming has been developed. Within the test study, the metrics which demonstrate performance of QoE are investigated, respectively including, initial delay (i.e., startup delay at the beginning of playback a video), frequency switches (i.e., number of times the quality is changed), accumulative video time (i.e., number and length of stalls), CPU usage, and battery energy consumption. Furthermore, the relation between effective parameters of QoS on the aforementioned metrics for different segment length is investigated. Experimental results show that the proposed virtualized system is agile, facility to install and use, costs less than real testbeds. As a result, the proposal system can be maintained adequate consuming CPU and battery energy consumption until 50% of launched machines during the process of experiments. Moreover, the subjective and objective performance study of QoE evaluation in the system has proved that the segment lengths of 6 to 8 seconds were faired and more efficient than others according to the investigated parameters.

Keywords: *HTTP Adaptive Streaming; Network virtualization; Testbeds, QoE assessment; QoS.*

I. Introduction

Modern day video streaming applications have developed significantly. The most ubiquitous technique to stream live video and video-on-demand is the Hypertext Transfer Protocol (HTTP). Adaptive streaming technology (HAS) takes advantage of HTTP protocol to make video streaming [1]. This advantage makes video streaming more popular in Internet applications because it utilizes HTTP as a primary means of transportation and it is easy to configure, firewall friendly and Network address translation (NAT). Several commercial vendors such as Netflix, Hulu, and YouTube use HAS technology to provide better QoE to end-users within limited Quality of Service (QoS) such as bandwidth, delay, packet loss and jitter of packets [2]. In HAS, the video content is encoded at several bitrates and each bitrate is packetized into small segments called chunks. Each chunk contains a fixed duration of video in seconds. The address of segments is stored in an Extensible Markup Language (XML) file called a media presentation description (MPD) or manifest file. Once the client reads the manifest file, it can easily stream the video content from a server or multiple servers in a Content Delivery Network (CDN). The benefit of the HAS is to allow a multimedia player to adaptively change the network conditions by switching between different representations of the same content. There are several companies that have their own implementation of adaptive multimedia streaming, such as Apple (HLS), Adobe (HDS), Microsoft (MSS), and MPEG-DASH as mentioned by [2; 3], most of them use similar mechanisms for adaptive streaming while applying different characteristics and formats.

Nowadays, The latest generation of the smart devices can provide higher computing capabilities, run software, and perform other tasks. Furthermore, they can use different network interfaces to connect to Internet [4]. Mobility as the key attribute of these devices allows end-users to consume multimedia service streaming. In addition, multimedia services are generating a high percentage of down streaming Internet traffic. For instance, the bandwidth usage of Netflix as a multimedia service provider climbs to nearly 37% of the Internet traffic and consumes more bandwidth than YouTube, Amazon, and Hulu. According to a report published by Cisco, global mobile data traffic between 2015 and 2020 will increase nearly eight times as much as current mobile traffic [5]. However, testbeds have been used to evaluate the performance of multimedia applications and provide a platform to fulfill testing requirements. Hence, many authors have proposed real testbeds and virtualized testbeds for conducting experiments. The comparative performance of delivering multimedia streams in both real and virtualized network testbeds has been investigated in [6]. They found the performance evaluation of the applied testbeds by monitoring the QoS parameters and the effect of the video transmission on these parameters. The different scenarios are revealed to show accurate comparison. Therefore, test results show that the packet loss and the delay of real-life scenario are consumed a higher rate than simulate scenarios.

Decision on selecting metrics to assess the performance evaluation of QoE and guarantee QoE over heterogeneous networks such as wire connection, wireless connection and mobile connection in real scenarios face the following issues:

1. Designing a real testbed to evaluate the performance of networks and heterogeneous video streaming applications is NOT cost-effective, mainly because collecting (or having full access to) the required resources for the testbed development can be a complex task.
2. Generating chunk size of the adaptive video streaming on the service providers to guarantee QoE is still under the indistinct decision, with regards to setting intra frames positioning in the group of picture (GOP).
3. Problems occur when a watching adaptive video streaming on a device that is connected to wideband or narrowband Internet wireless connections.
4. Characteristic of the video regarding segment length, which a factor leads for utilizing high CPU usage and battery energy consumption of heterogeneous devices when delivering and rendering the video content on the client-side under different QoS.

Therefore, the key contribution of this paper is to design a generic testbed by virtualizing heterogeneous components with a different functionality in a physical node. In order to evaluate the performance of multimedia applications, the following parameters have been studied to reduce the cost of experimental tests, time to apply, and agility.

- Selection of an optimal position of intra frames in the GOP when generating video content on the service providers for wireless customers in order to provide better QoE.
- Investigating the useful metrics, which decide on QoE of end-users, objective metric such as initial delay, instability of quality, buffering underrun, and buffering length and a subject metric.
- Find the correlation between QoS parameters which impact on the evaluation of QoE metrics for different segment length.
- Resource consumption metrics such as CPU usage and battery energy consumption are investigated for the proposal system and the clients are used multimedia application.

The rest of the paper is structured as follows. Section II presents the background of testbeds for delivering multimedia content. Section III explains the proposed system and the select of QoE parameters to evaluate subjective and objective QoE performance. Section IV shows the performance study and results. In section V, a conclusions and future works are presented.

II. Related Work

In this section, a background for the testbeds and adaptive video streaming is presented and some testbeds designed for multimedia applications are reviewed.

Most of the traditional websites are becoming multimedia websites. Moreover, streaming multimedia content over HTTP is a combination of downloading and concurrent playback. It transmits the video data to the end-users via HTTP where it is stored in a multimedia application buffer. When a sufficient amount of data has been downloaded, the client can start to play the video from the buffer. Furthermore, the increased trend of smart devices to playback multimedia content in heterogeneous networks makes the need of designing testbeds and experiments in order to evaluate their performance. However, designing testbeds by using the virtualization technique has become a key component of the network testbeds because they allow testing protocols and applications varying the network conditions. Generally, network testbeds have been used to define specific experiments [5]. Once a testbed has been defined with its resources for a specific experiment, it can be used to test the performance and gather measurements to analyze the results. But what benefits the research and industry community most is to have testbeds that allow the performance of several types of experiments. Therefore, a wide range of testbeds such as: OneLab, Emulab, G-Lab, NetKit, and PlanetLab have been designed for different goals. Nevertheless, it is quite difficult for these testbeds to reserve enough resources for their own experiments and they have to meet different requirements than those of testing applications [7]. Moreover, virtualizations systems are utilized to reduce the cost of the testbed setup because the hardware resources are reduced and they allow for the creation of more complex network scenarios. In [8], authors proposed a real characterized testbed of DASH video streaming to measure the impact on the user experience in order to investigate the initial delay, stall, throughput, and CPU consumption for different segment sizes. Furthermore, in [9], the implementation of a real-time test platform for adaptive video streaming has been proposed. The physical devices are connected to the simulated network in order to evaluate the performance of the requests for different segment size of adaptive video streaming.

Service providers are always trying to guarantee QoE. Multimedia providers interest the evaluation of QoE in order to understand on selecting which video content is ready for the end-users. Meanwhile, selection number of representations (video quality), optimal video encoding, and length of encoded chunks, are key issue to prepare the video content for the multimedia service providers. Moreover, authors of [2] described the segments needed to be long enough to allow high coding efficiency of the source video encoder. In [10], authors deduced that the segment length needs to be short enough to allow for a fast reaction in high network fluctuation. Both requirements need to be considered by content providers during the preparation of adaptive video streaming contents. In [11], authors proposed that a YouTube-DASH content provider with

a segment interval of 2 seconds could obtain 95% reduction in bandwidth for low-quality videos and up to 83% reduction for HD videos in cases where users do not watch the videos completely. Furthermore, MSS, HLS, and HDS proposed different segment length. Segment length in MPEG-DASH and 3GP DASH has not been specified. HbbTV also supports the high range of segment length. Finally, Bitmovin depicted the optimal selection for a segment length: between 2 and 4 seconds.

In HTTP adaptive video streaming, which segment may be downloaded, when to start the download, and the client adaptation algorithm decides how to manage the buffer video size. The initial delay depends on the available transmission data rate and the encoder settings. Authors of [12] showed that the initial delay depends on the bitrate of the chunks that are downloaded to fill the buffer at the beginning of the streaming. If chunks are downloaded in high quality during the startup phase, the initial delay will be longer, but the video will begin to stream in high quality. [13; 14] explained that the impact of the initial delay strongly depends on the user applications. Moreover, authors of [13] established that high initial delays reduce the perceived quality. When the throughput of the video streaming application is lower than the lower video bit rate, the application buffer will stall. Additionally, QoE is influenced by the duration and the frequency of stalling events. The escalated duration of stalling decreases the quality of video streaming. This suggests that one long stalling event is preferable over short, but frequent stalling. However, the position of stalling is not crucial. The authors of [2] described the analyzed high compression for both video stalling and playback videos. They showed that stalling is worse than playback videos encoded in high compression and an inaccurate adaptive algorithm in HAS degraded QoE.

The approach used in this paper is different from other works have been proposed above, the virtualized testbed is proposed for evaluating QoE of adaptive video streaming. The proposal testbed can be launched heterogeneous network scenarios such as wireless scenario and CDN scenario. Therefore, the QoE evaluation in the system based on subjective and objective approaches in order to deploy a decision to generate optimal segment length by service providers for wireless end-users.

III. System Description

In this section, an overview of the proposed system is explained and reveals more detail of the components. Then the parameters that are used to evaluate the QoE of end-users and the implementation model of the proposed system are described.

A. System overview

Generally, the transportation of multimedia contents over the Internet can be considered using vast networks. In these network scenarios, there is the appearance of CDNs, user connect to ISP through wire connection and wireless environments such mobile and Wi-Fi. The content of video

is delivered from the cache servers to the end-users through different network access. In the case, where the content is not available from correspond replica server. The CDN system redirects the client request to a place where is hold the copy of the content, the redirection may proceed to other replica or service provider [5]. The process of delivering adaptive media streaming in Internet is shown in figure 1.

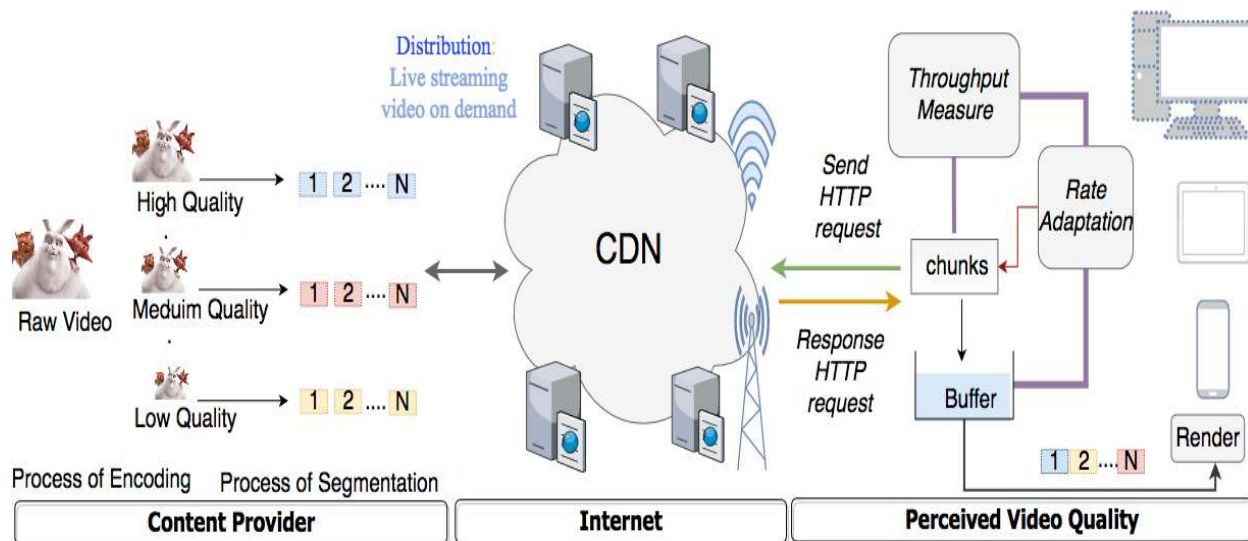


Fig. 1. System overview.

The principal system design of virtualized nodes in the testbed is grouped into parts; virtualized network nodes and virtualized host nodes. Availability host nodes can be a server and a client, who serves a resource to a client, is defined as a server and who receive a resource from a server is defined as a client.

The network nodes can be routers and switches, network virtual machines are appeared with different functionality. The routers operate routing at the network layer of the OSI mode and responsible for forwarding packets among the nodes. Therefore, the developed part of the system includes application-programming interfaces (API). Resources can be available in any node of the system. The hosts have API to share resources and present different functionality. Nodes in the system are virtualized over a Linux Operating System, using Linux is to avoid restrictions still exist in other testbeds in delivery adaptive video streaming to various end-users. Also, the design easily allows measuring various experiments through monitoring equipment.

The topology of the system includes the multiple virtual machine (VMs) components such as: a main server, cache servers and network traffic shapers (virtualized networks), all of them are operating from a single raw physical machine (PC). These VMs connect together in a different range of network classes because different networks can provide facility configuration and avoid conflicts between connections. More specifically, virtual bridges provide bridge connection between the virtual components and the physical device. The topology of the virtualized system is shown in figure 2.

The main server is the multimedia content service provider and selects the codec and the segmentation of the video content. This server serves end-users through providing multimedia streams.

The cache server is the mirror of the content provider. It provides caching video content in order to reduce the clients' requests by hitting contents. In addition to this, the network traffic shaping emulator is used to shape available downstream and upstream networking by prioritizing network resources and guaranteeing certain bandwidth based on predefined policy rules. It uses concepts of traffic classification, policy rules, queue disciplines and QoS. The shaping of the network is a combination of traffic control (TC), queuing discipline hierarchy, Token Bucket (HTB) and Network Emulation (NetEm). This is done in order to shape and control the network's uplink and downlink, delay, and packet loss ratio.

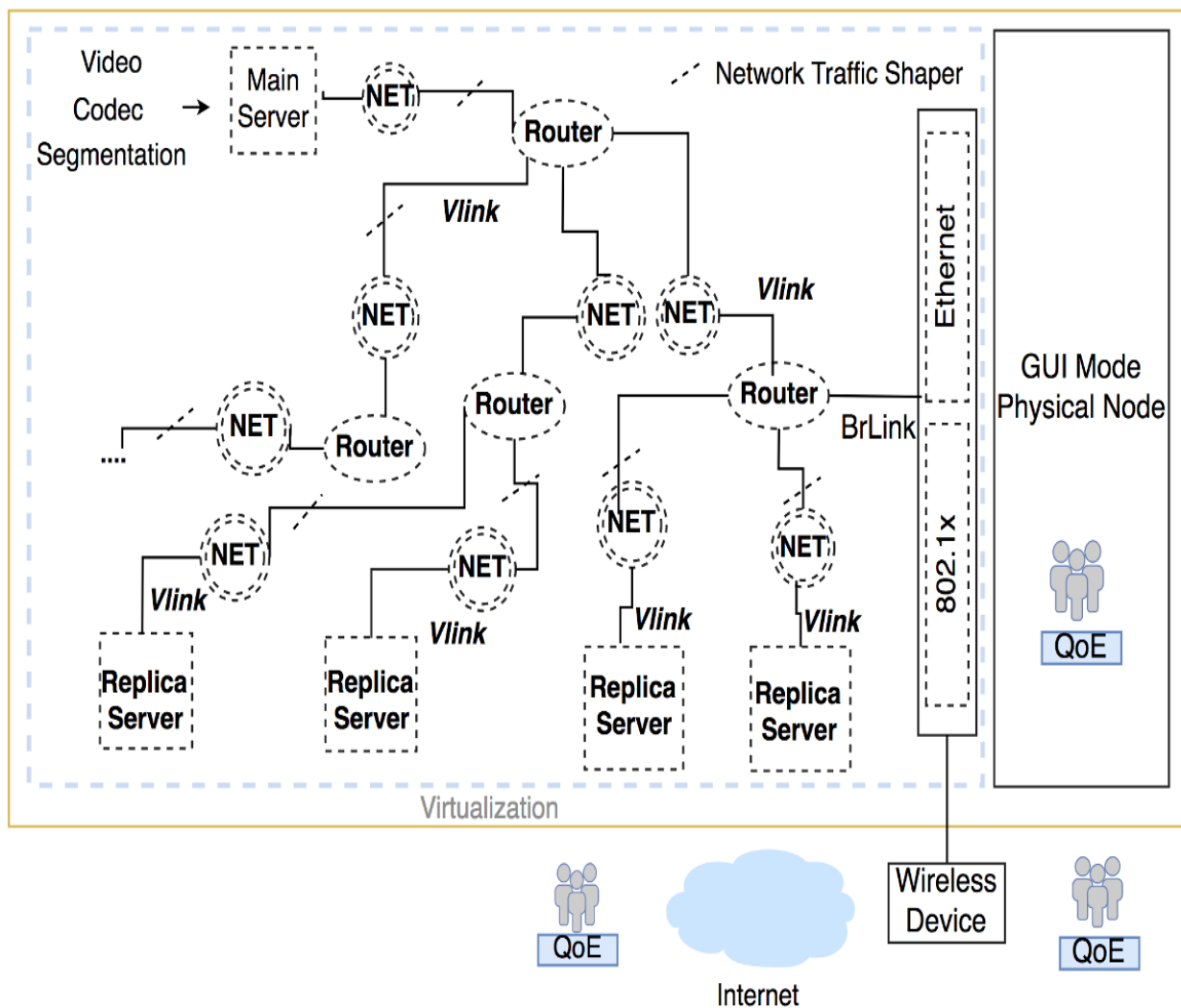


Fig. 2. The topology of the proposed system.

The proposed testbed system has the ability to connect to external networks; intranet and Internet. The clients may be heterogeneous devices such as virtualized machine, laptops, mobiles, and PCs. The end-user devices may available under different operating systems. The process of the proposed system operation is shown in figure 3 (a), when the system starts to operate, the VMs behaviours on the physical machine start to run. The behavior of content provider is to provide video content with different segment length. The cache server is ready to hold a copy of the requested content to provide future caching content to end-users. Network traffic shaper shapes or restricts automatically the downstream and upstream of the available throughput. Both internal and external end-users through the wireless behaviour are connected to the system to be able to share the resources.

The other part in the system, using APIs in order to share the resources, the API is resided in the main server these resources can be shared among other hosts by accessing to the main server. Other hosts as defined as clients or end-users are able to receive the shared resource. The API is used to send/receive data among the hosts and interchange information in order to the main server knows about the situation of the client session. The API allows the end-user to demonstrate a feedback on the service sent by the main server. In order to assess the adaptive video streaming as shared recourse by the API. The API is developed to provide evaluation approach in order to assess delivering service to the end-users and the API on client-sides notify the server in the term of sending back a statistic feedback. The metrics are used to evaluate the QoE of the end-users will be shown in next subsection.

B. QoE Metrics

In adaptive video streaming, subjective and objective are important approaches can be used to assess the delivered video. Indeed, there are two primary aspects to assess video quality, display quality (fidelity), image quality is sufficient for the device's screen size. And transport quality (startup time, buffering and stalling), how long does the video take to start, and does it play smoothly. The assessment model is grouped into parts an objective assessment and a subjective assessment. The following QoE metrics are used to provide the assessment model in our system is to evaluate QoE regards to satisfactory and annoyance of the service, as depicted in figure 3 (b). Therefore, the detail of each metric will be defined and explained clearly next.

- *Initial delay and buffer length:*

The client application creates a buffer length for Y duration in seconds, the value of Y based on video segment length and maximum availability of buffer size related to QoS factors such as high bandwidth variation and packet loss rate. Buf_sl_i denoted as buffer size of segment length, where $Buf_sl_i < Buf_sl_{i+1}$, $i = 1$ to M where N and M is maximum number of segment length. The client application at time T_0 starts to receive data and store in buffer application before being played back at time T_n , which called the initial delay or startup time of the video as shown in figure 4.

- *Number of switches (number of times the video quality is changed):*

The number of quality switch describes as oscillation of the video session. It depends on the number of bits flow through the channel per seconds and instability of throughput. High value of frequent switching leads to decrease the QoE as described in the following formula .

$$\sigma^2 = \sum_{i=0}^n g(s_i) \quad \begin{cases} 1 & \text{if } i = 1 \\ 1 & \text{if } f(s_{i-1}) \neq f(s_i) \\ 0 & \text{else} \end{cases} \quad (1)$$

Where, s_i segment _{i} $i \in [0 - N]$... Segment Index f_{s_i} ... bitrate of segment i

- *Video accumulative time:*

Accumulative video time metric represented as exceed video length than default video length, insufficient bandwidth cause to buffer of video application underrun and this parameter impact of the QoE end-user that related to number of stalls and stall duration, Where

$$\text{accumulative time} = \text{Initial delay} + \gamma \quad (2)$$

Where, $\gamma = \sum_{i=0}^n Z_i$

Z_i denoted as duration of each stall, where i number of stalls may 1,2, ... N.

- *DMOS (Difference Mean opinion score)*

The DMOS is therefore the difference between the MOS value obtained for the original video and the MOS value obtained for the delivered video. So, DMOS value gives the mean subjective value of the difference between the original and the delivered one. A value of 0 means no subjective difference found between the video by all the viewers [X].

In order to find the threshold values in the proposal model. We take benefit of the subjective assessment. We establish a mapping relation between subjective and objective in order to automatically assess the QoE. Three important parameters are taken to evaluate the QoE such as initial delay of the video, switch between qualities and video stalling. These parameters are massively annoyed end-users. Figure (5) depicted the reverse relation between the objective metrics and subjective assessment. Each value of subjective assessment is mapped to the objective metrics and this is led to know which values are annoyed and satisfactory the end-users.

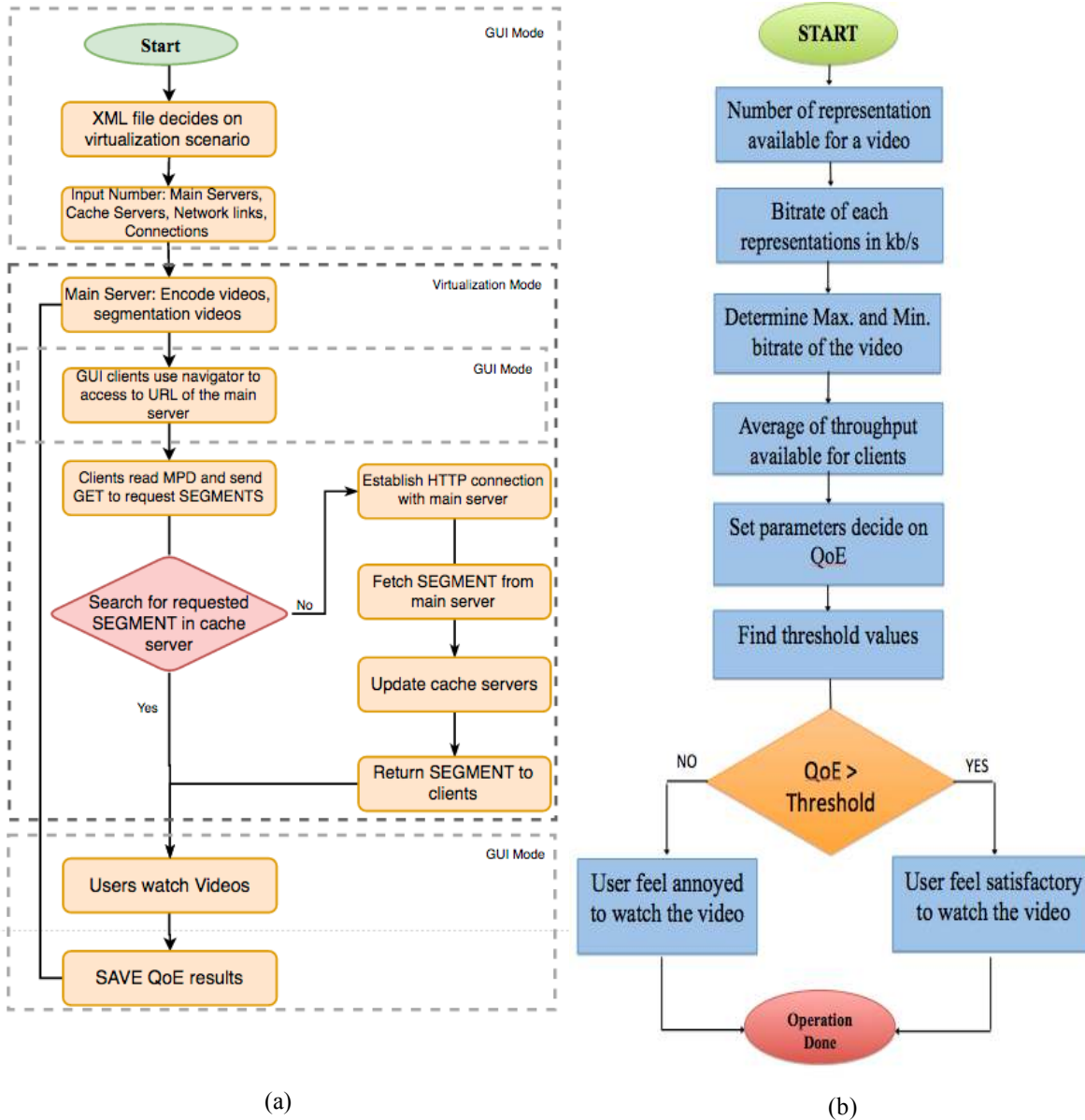


Fig. 3. System algorithm: (a) Operation functional view, (b) QoE evaluation.

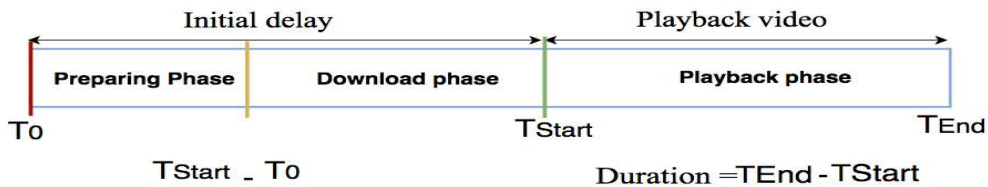


Fig. 4. Application's buffer size and process of playback video.

Therefore, the threshold value is found from the base of the subjective evaluation, also, the range of the subjective evaluations is 1 to 5. The output of the subjective evaluation will be a positive where a group of users give a positive feed back on the received video and its range can be 3 to 5, otherwise, the range 1 and 2 is indicated as a negative feedback on the perceived quality. The median value is set as threshold to reveal the decision on satisfactory and annoyance of the end-users.

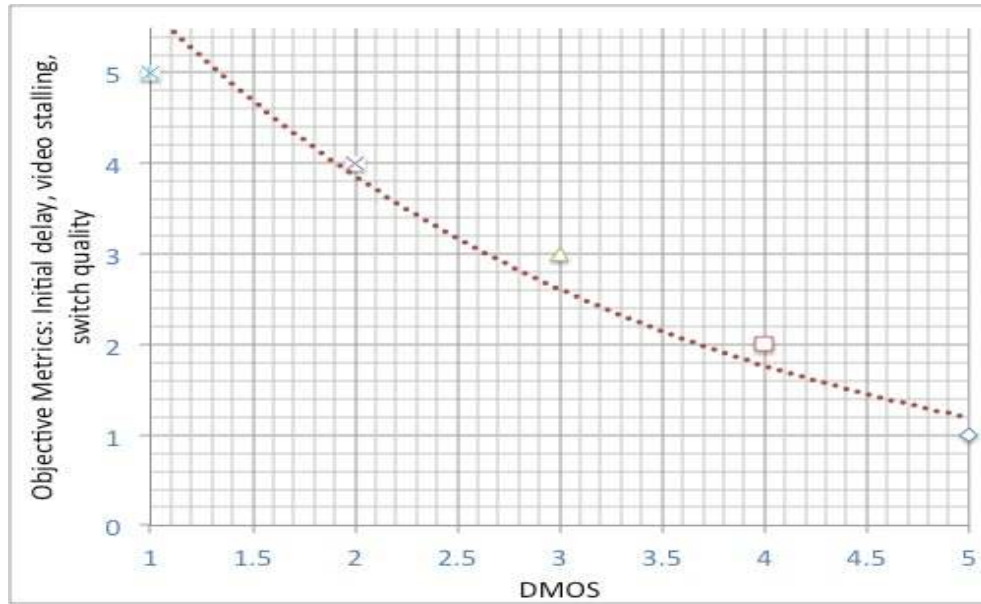


Fig. 5. Relation between objective metrics and perceptual evaluation.

C. Resource usage parameters

CPU usage and battery energy consumption for the VMs and the entire virtualized system are measured by using the Linux monitoring tools. In the VMs, The number of cycles executed per second is analyzed when a user watches a video for different segment lengths. The average of the CPU usages will be calculated which influenced by some factors such as: high encoded video quality, quality video switching, CPU environment and the behaviour of the video playback application and availability of clients' throughput

Moreover, battery energy consumption measurement includes real-time monitoring power usage of the virtual devices during the playing back videos. The test with varying available throughput and automatic bitrate adaptation will be used to define how much energy will be consumed during the oscillation of video quality for different segment size. The energy impact will be observed while the reaming charge is 100 percent. Thus, both aforementioned parameters will be found for the entire proposed system.

D. Proposed Model Implementation

In order to implement the proposed design, the VNX program, which parses the scenario description over a Linux machine [5], is used to virtualize the nodes and the networks. In this program the XML language allowed describing the virtual network scenario. In our scenario, the video content provider is deployed on HTTP web server, which is NGINX server version 1.8.1, with Linux-Ubuntu 14.10. Squid caching proxy that supports HTTP web has been installed on the cache machines, which is the Linux-Ubuntu 14.10 operating system. The network traffic shaper operates in Linux-Ubuntu 14.10. It includes the shell script that automatically emulates the network clients' throughput links according to the bandwidth file. Moreover, the physical wireless access point is a dual-radio 802.11n with supporting high data rate.

Dynamic adaptive streaming over HTTP based technology on JavaScript (DASHJS) is an open source application, which has been used as a part of our implementation to serve the clients' video streaming application.

The network test tools have been used in the testbed such as Ping to show connectivity of the links and Iperf to measure the QoS parameters, as it reports the bandwidth, delay and packet loss. Moreover, we modified the open source application to record activity of objective metrics and recorded information sends to the main server over HTTP protocol. This process is done by using a PHP program, record the track of the user such as IP address and after finishing a play back of the video, all information of metrics are transported to main server to assess the QoE in real time.

IV. Performance Test Study

A. Testbed equipment

In order to demonstrate the usefulness of the virtualized network testbed upon the implementation adaptive multimedia streaming, it is necessary to evaluate the performance of the testbed about generic which supporting heterogeneous multimedia applications, physical size, mobility, scalability, realism of the tests and cost-effectiveness.

B. QoE parameters evaluation

The Big Buck Bunny raw video is used to perform the QoE experiments in the proposed testbed system. In order to provide DASH content with varying bitrates, 18000 frames (60 fps) are encoded with Lib-X264 and GPAC-MP4Box. We provide 20 representations that consist of three levels of video quality: low, medium, and high definition quality. Each level of quality provides different bitrates, which are sequenced from low quality, which has a resolution of 320x240 with a bitrate of 50 kbps (baseline profile), to the ultimate quality, which has a resolution 1920x1080 with a bitrate 5000 kbps (high definition profile). In fact, we provide 7 versions of the video with a different segment duration starting with a segment length of 1

second and then 2s, 4s, 6s, 8s, 10s and ending up with 15s. Therefore, to conduct these experiments, the output data have been collected from aforementioned parameters.

In the first experiment, the initial delay varied in the experiment, which depended on the segment duration. This fact is shown in figure 6 (a). The initial delay increased when the segment length increased. Generally, the tendency of these lines is to be uniformly related to the increase of segment length. Segment with a duration of 4 seconds appeared as inflection point associated with the others. Moreover, few seconds of unrestricted bandwidth led to reduce the initial delay of the playback video in large segment durations compared to short segments as shown in green line of figure 6 (a).

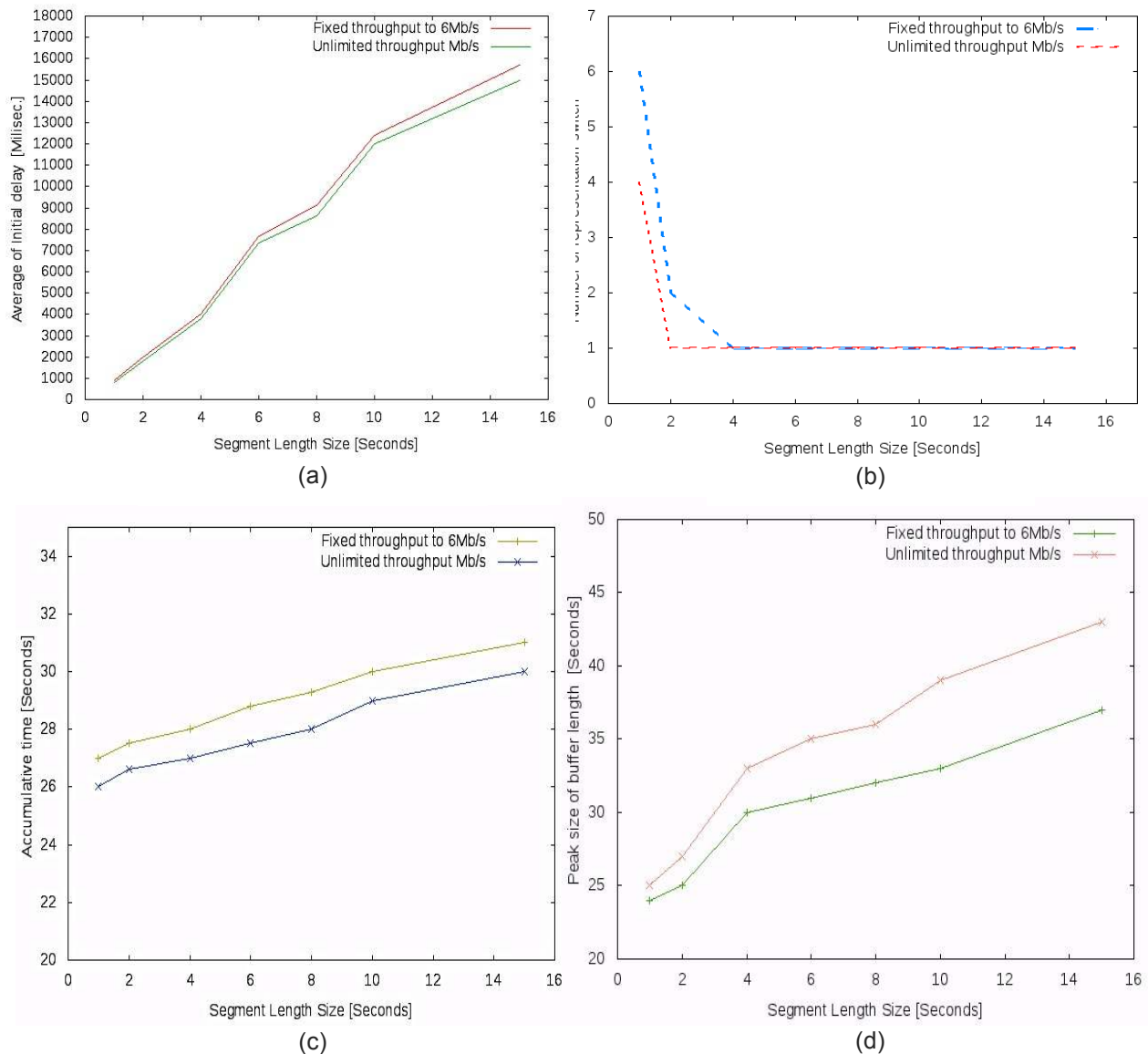


Fig. 6. QoE metrics: a) Initial delay, b) Quality oscillation, c) Accumulative time, d) Maximum buffer size

In the second test, two different tests are performed with the system as the previous experiment. The goal was to evaluate the number of switches (present different quality). In those tests, a relationship between the different numbers of switches before 4 seconds segment durations was observed, as shown in figure 6 (b). The blue line represents the first test results and the red line represents the second one which quality oscillation has depicted 4 and 6 times with a second segment duration after the decreased tendency of the line reached the 4 seconds segment duration. From 4 seconds, segment duration onwards the lines coincided until the end of the video.

In the same manner, tests for accumulative video time in the third experiment were applied. The 4 seconds segment duration appears to be the inflection point between pitch curves with a low tendency curve as shown in figure 6 (c). From the results, it can be noted that increasing the segment duration for both available network throughputs causes increment of the accumulative time. The accumulative time curve rises up rapidly when the short segment duration is being used. Meanwhile, ascending the curve slowly above 4 seconds duration means that the accumulative time of short segments is 35% higher than long segments. Figure 6 (d) depicts the results described in the fourth experiment, which is performed to measure the effect of the buffer length. The curves show that, after 4 seconds segment length, the higher buffer size is used.

In the subjective evaluation, in order to obtain the DMOS value we were consulted to 10 video evaluation experts in the university. The assessment process of DMOS is classified into groups; DMOS for initial delay and DMOS for video stalling and switch quality. To carry out the process of DMOS the original video and the delivered video are shown to the experts. The rating scale is indicated from 1 to 5, the results of the subjective evaluation are presented in table 1. Also, ratio of assessing the small segments is become worse than the large segments otherwise the initial delay of large segments present worse case of evaluation.

Table 1. Evaluation QoE based on a subjective and an objective method.

Segment	Initial Delay (Sec.)		Switch quality (Times)		Stalling video (Times)		DMOS Initial Delay		DMOS Switch & Stalling	
	Restricted	Unrestricted	Restricted	Unrestricted	Restricted	Unrestricted	Restricted	Unrestricted	Restricted	Unrestricted
1	1,1	1,01	6	4	7	5	4	4	1	2
2	2,4	2,04	2	1	3	2	4	4	2	2
4	4,1	4,02	1	1	3	2	3	4	3	3
6	7,8	6,89	1	1	3	2	3	4	4	3
8	9,100	8,79	1	1	3	2	3	4	4	4
10	15,50	10,3	1	1	3	2	2	3	4	4
15	16,01	15,2	1	1	3	2	2	2	4	4

C. Resource usage evaluation

The CPU usage was also studied with this testbed. In this experiment, average of CPU load is captured for 300 seconds for segment lengths 1, 2, 4, 6, 8, 10 and 15 seconds. Results are plotted in figure 7 (a). From the obtained results one can observe the following:

- Short segment duration uses shorter peak of CPU usage to request a segment than the increasing segment duration.
- Short segments consume upper bound of CPU during decoding the video. This effect is the opposite in longer segments.

The rate of the CPU load is decreased in the different network environments (by 60% from one to four segment duration). From 4 to 15 seconds of segment duration the rate is decreased by 8%. The average energy consumption is presented in figure 7 (b). Rate of the consumption in small segments is higher than the large segments for both network connections, which can be estimated 30% higher. Retrieving small segments establish many connections to download into buffer's application makes the device consuming higher energy than large segments.

Finally, the quantify impact of for resources of CPU usage and battery energy consumption by virtualization of the devices during the experiments are monitored on the same physical hardware as shown in figure 8. In the system, the number of virtualized devices is affected on both CPU time and the energy consumption. It is observed when the device number equals up to 25, the CPU metric values are dramatically increased by 50%, up to 16 VMs consumes higher energy. The proposed testbed is operated for the optimal number of VMs = 25 through running the experiments and evaluate the QoE of multimedia application.

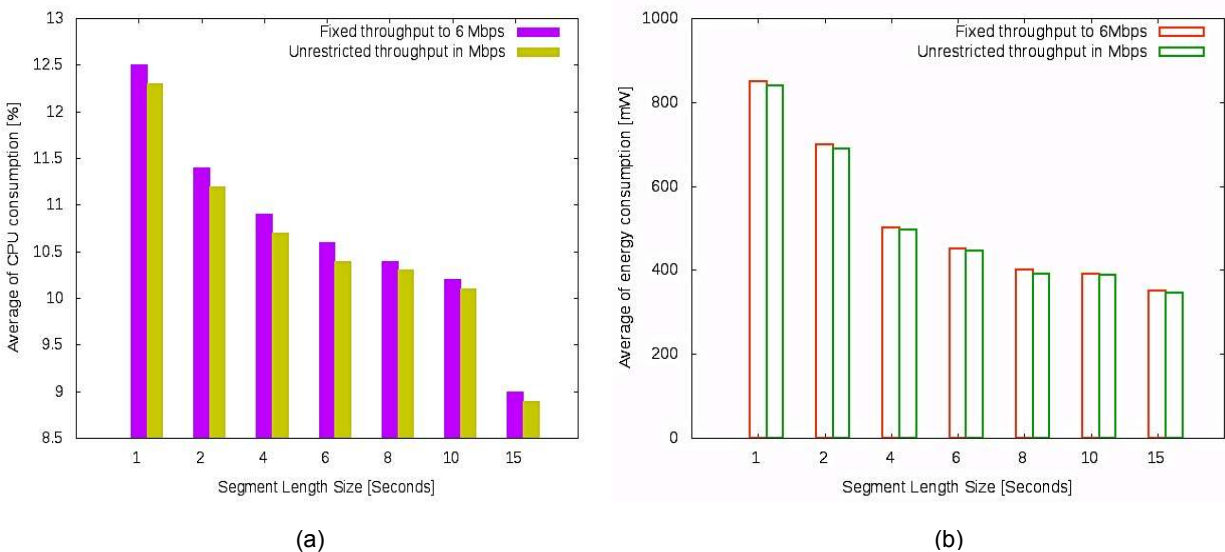


Fig. 7. Consumption metrics: (a) CPU usage of a client, (b) Energy consumption of a client.

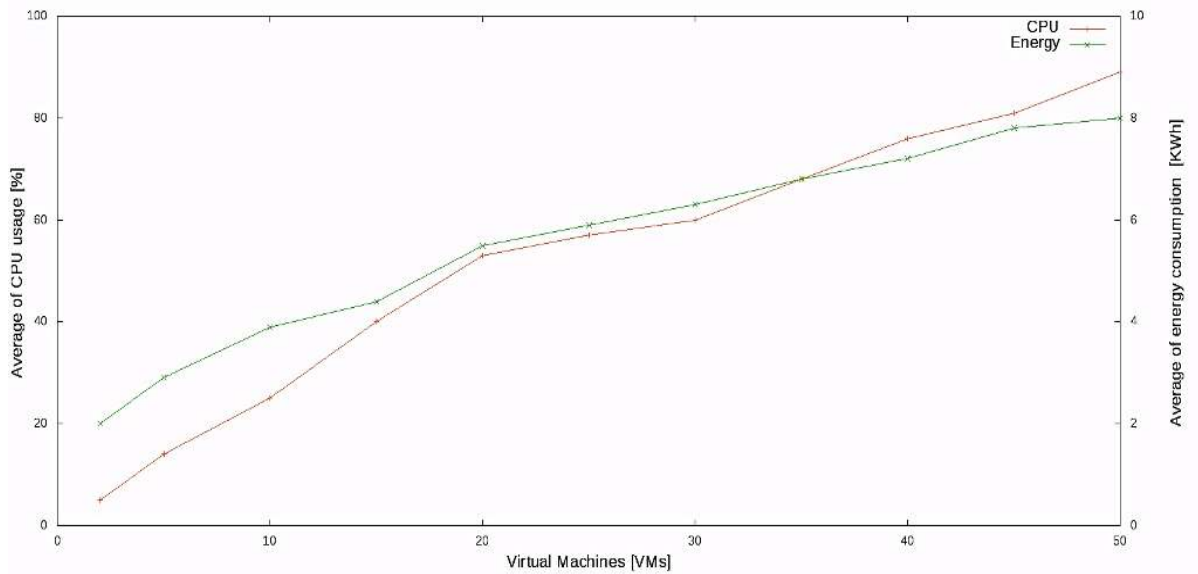


Fig. 8. System consumption metrics.

D. Proposed Approach Comparison

In this section, benchmark parameters such as physical size, mobility, scalability, realism of the tests, cost-effectiveness, network simulation and emulation, core network and etc. between our proposed testbed scheme and other testbeds are shown. The impact of factors on the user experience according to the performed tests, which video segment length provides better QoE, are also presented. Table 2 shows the similarity and dissimilarity of the network testbeds parameters.

V. Conclusion

This paper has proposed a virtualized testbed design to conduct real time adaptive video streaming experiments in different network environments. The proposed system presented a virtualized complex network scenario and various virtualized nodes with different functionality. The system testbed has been used to evaluate QoE of adaptive multimedia. The evaluation approach is based on subjective and objective. To select the process of the QoE evaluation, some objective metrics and a subjective metric are presented to demonstrate accurate decision on the perceptual video. In the system to arrive the evaluating QoE, adaptive video content prepared in different segment length and two networks stratagem throughputs are set up.

Table 2. Benchmark comparison between proposed approach and conventional approaches.

Testbed name	Simulate testbed [7]	Real testbed [8]	Simulate testbed [9]	Simulate testbed [13]	Flamingo [15]	Proposed testbed
Tested physical	1 Node	5 Nodes	1 Node	1 Node	1 Node	1 Node
Mobility	Easy	Low	Easy	Easy	Easy	Easy
Scalability	High	Low	Medium	High	High	High
Cost efficiency and dollar-cost per device	Medium	High	Low	Medium	Medium	Medium
QoE Method based on	-	-	-	-	-	Decision algorithm.
QoE Metrics	Only video quality oscillation	Initial delay, video stalls, switch frequency	CPU, TCP.	Video stalls, buffer size, switch frequency, MOS	QoE based only on (MOS)	Initial delay, buffer length, video stalls, switch frequency, DMOS, CPU and energy
Realism	Combine of Simulation and emulation	%100 Real	Simulation (Math. Models)	Simulation (Math. Models)	Combine of Simulation and emulation	Emulation (Live network)
Topology of the core network (routers and switches) based on	Simulation	Real	Simulation	Simulation	Simulation	Emulation
Cache service	-	-	-	-	Support	Support
Machine Learning-based Virtual Network	-	-	-	-	Support	No defined yet
Preferable segment length	Not defined	Small segments	Not defined	Not defined	Not defined	6-8 seconds duration
Language used for describing the simulation	Python	-	-	-	Python	XML
VMs Operating system based	Linux	-	-	-	Linux	Linux
VMs Chain distribution	-	-	-	-	-	Support
Network topology presented for experiments	Simple	Simple	Simple	Simple	Complex	Complex
Description	Virtualized network testbed to perform experiments by Mininet	Real nodes to design real testbed to experiment the QoE	Simulate Platform for mobile video streaming using OPNET.	Simulate Platform for LTE-mobile video streaming using ns-3	Virtualized network testbed to perform experiments by Mininet	Virtualized network testbed to perform experiments and run several virtual nodes on a single physical machine

An important benefit of the schematic virtualized design was the economic cost compared to real existing physical devices and the scenario can support 25 VMs for maintaining the consumption of optimal battery energy and CPU usage for the real time experiments. The proposed system offered the possibility of applying different operating systems, ease in implementation and applying models and agile.

Moreover, from the relation between the consumption metrics and subjective and objective metrics the evaluation of the observed QoE of the end-users showed the optimum segments' duration to feed the server was from 6 to 8.

For future work, a sophisticated design based on machine learning will be implemented in order to carry out more complex experiments. In addition, an intelligent method will be proposed to evaluate the performance of QoE.

Acknowledgment

This work has been supported by the “Ministerio de Economía y Competitividad”, through the “Convocatoria 2014. Proyectos I+D - Programa Estatal de Investigación Científica y Técnica de Excelencia” in the “Subprograma Estatal de Generación de Conocimiento”, project TIN2014-57991-C3-1-P

References

- [1] Ozgur Oyman, Sarabjot Singh. "Quality of experience for HTTP adaptive streaming services." *IEEE Communications Magazine*, vol. 50, no. 4, pp. 20-27, April 2012.
- [2] Michael Seufert, Sebastian Egger, Martin Slanina, Thomas Zinner, Tobias Hobfeld, Phuoc Tran-Gia. "A survey on quality of experience of http adaptive streaming." *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp.469-492, 2015.
- [3] Laura García, Jaime Lloret, Carlos Turro, Miran Taha. “QoE assesment of MPEG-DASH in polimedia eLearning System.” 5th International Conference on Computing, Communications and Informatics, India, September 2016.
- [4] Jaime Lloret, Miguel Garcia, Marcelo Atenas, Alejandro Canovas. “A QoE management system to improve the IPTV network.” *International Journal of Communication Systems*, vol. 24, no. 1, pp.118-138, 2011.
- [5] Miran Taha. "A Novel CDN Testbed for Fast Deploying HTTP Adaptive Video Streaming." *Proceedings of the 9th EAI International Conference on Mobile Multimedia Communications*, ACM, pp. 65-71, June 2016.
- [6] Jimenez, Jose M, José Oscar Romero Martínez, Albert Rego, Avinash Dilendra, and Jaime Lloret. "Study of multimedia delivery over software defined networks." In *Network Protocols and Algorithms*, vol. 7, no. 4, pp. 37-62, 2015.

[7] Anatoliy Zabrovskiy, Evgeny Kuzmin, Evgeny Petrov, and Mikhail Fomichev. "Emulation of dynamic adaptive streaming over HTTP with Mininet.", proceeding of the 18th conference of fruct association, pp. 391-396. 2016.

[8] Yao Li, Sujit Dey, Don Gillies, Faith Ulupinar, Michael Luby. "User experience modeling for DASH video." 20th International Packet Video Workshop, In Packet Video Workshop, IEEE, pp. 1-8, December 2013.

[9] Yomna M. Hassan, Ahmed Helmy, Mohamed M. Rehan. "Effect of varying segment size on DASH streaming quality for mobile user." Engineering and Technology (ICET), 2014 International Conference on, pp. 1-4, April 2014.

[10] Jaehyun Hwang, Junghwan Lee, Chuck Yoo, "Eliminating bandwidth estimation from adaptive video streaming in wireless networks." Signal Processing: Image Communication, Vol. 47, pp. 242-251, September 2016

[11] Dilip Kumar Krishnappa, Divyashri Bhat, Michael Zink. "DASHing YouTube: An analysis of using DASH in YouTube video service." In Local Computer Networks (LCN), IEEE 38th Conference on, IEEE, pp. 407-415, October 2013.

[12] Truong Cong Thang, Hung T. Le, Hoc X. Nguyen, Anh T. Pham, Jung Won Kang and Yong Man Ro. "Adaptive video streaming over HTTP with dynamic resource estimation." Journal of Communications and Networks, Vol. 15, no. 6 pp. 635-644, 2013.

[13] Florian Metzger, Christoph Steindl, and Tobias Hoßfeld. "A Simulation Framework for Evaluating the QoS and QoE of TCP-based Streaming in an LTE Network." In Teletraffic Congress (ITC 27), 27th International, pp. 168-176, 2015.

[14] Tobias Hossfeld, S. Egger, R. Schatz, M. Fiedler, K. Masuch and C. Lorentzen. "Initial delay vs. interruptions: Between the devil and the deep blue sea." In Quality of Multimedia Experience, Fourth International Workshop on, IEEE, pp. 1-6, July 2012.

[15] Flamingo Project. Available at:

http://cordis.europa.eu/docs/projects/cnect/8/318488/080/deliverables/001318488FLAMI_NGO_62.pdf