Research Article

Real-Time QoE Monitoring System for Video Streaming Services with Adaptive Media Playout

Mingfu Li [iD],[1,2] Chien-Lin Yeh,[1] and Shao-Yu Lu[1]

[1]*Department of Electrical Engineering, School of Electrical and Computer Engineering, College of Engineering, Chang Gung University, Guishan District, Taoyuan City 33302, Taiwan*
[2]*Neuroscience Research Center, Chang Gung Memorial Hospital, Linkou, Guishan District, Taoyuan City 33305, Taiwan*

Correspondence should be addressed to Mingfu Li; mfli@mail.cgu.edu.tw

Quality of Experience (QoE) of video streaming services has been attracting more and more attention recently. Therefore, in this work we designed and implemented a real-time QoE monitoring system for streaming services with Adaptive Media Playout (AMP), which was implemented into the VideoLAN Client (VLC) media player to dynamically adjust the playout rate of videos according to the buffer fullness of the client buffer. The QoE monitoring system reports the QoE of streaming services in real time so that network/content providers can monitor the qualities of their services and resolve troubles immediately whenever their subscribers encounter them. Several experiments including wired and wireless streaming were conducted to show the effectiveness of the implemented AMP and QoE monitoring system. Experimental results demonstrate that AMP significantly improves the QoE of streaming services according to the Mean Opinion Score (MOS) estimated by our developed program. Additionally, some challenging issues in wireless streaming have been easily identified using the developed QoE monitoring system.

1. Introduction

Internet Protocol Television (IPTV) services [1] have been becoming increasingly popular among telecommunication companies. Network/service providers are concerned with not only the cost of constructing an IPTV service system [2] but also its service quality [3, 4]. In the past, Quality of Service (QoS) of network services is the main concern. However, QoS cannot accurately characterize the users' perception. Therefore, the concept of Quality of Experience (QoE) was proposed and studied recently [5–13].

The most accurate method to obtain QoE is doing subjective tests [14] that are time-consuming and costly. Therefore, we proposed a cost-effective and real-time QoE evaluation method in [5], where a hybrid QoE assessment scheme was employed. In the hybrid QoE assessment, subjective tests only need be performed occasionally for updating the parameters of QoS-QoE mapping functions. Hence, the cost of a hybrid QoE assessment scheme reduces significantly. In [7], the QoE measurement for P2P-based IPTV services was studied. Another paper [8] proposed a QoS/QoE

mapping and adjustment model for the cloud-based multimedia infrastructure. Additionally, since streaming services over wireless networks have been becoming popular, the QoE enhancement algorithms, protocols, and evaluation methods for wireless streaming services were proposed [9–12]. In [13], Chen et al. presented a comprehensive survey of the evolution of video quality assessment methods and analyzed their characteristics, advantages, and drawbacks.

Since the QoE evaluation for streaming services becomes more and more important, several QoE evaluation or monitoring tools were developed [15–19]. A QoE monitoring tool for multimedia quality assessment in NS-3 network simulator was presented in [15]. However, since the objective and full-reference QoE assessment method is used in [15], it is costly and cannot accurately indicate the users' perception in a real commercial video streaming system. The paper [16] developed a testbed, which allows obtaining conclusive results regarding QoE in video-mediated group communication, for controlled experiments. In [17], Jeong and Ahn designed the QoE monitoring function for video services by extending ITU-T IPTV QoS/QoE metrics and integrating to OMA-DM

protocol agent. Although the monitoring service can improve the user experience, the presented QoE factors are still not good enough. In [18], the proposed concept, Monitoring of Audio Visual Quality by Key Performance Indicators (MOAVI), is able to isolate and focus investigation, set up algorithms, increase the monitoring period, and guarantee better prediction of perceptual quality. In [19], Ickin et al. added functionality to the VideoLAN Client (VLC) media player [20] to record a set of metrics from the user interface, application-level, network-level, and the available sensors of the device. Based on these metrics, the proposed VLQoE, a QoE instrumentation for video streaming, can infer the QoE of streaming services on a smartphone.

To improve the playout quality of video streaming services, Adaptive Media Playout (AMP) schemes [21, 22] have been used. In this work, an AMP playout system is implemented into the VLC media player, which is a free and open source cross-platform multimedia player. Since the playout rate of the AMP system can be dynamically adjusted during video playback, the effects of AMP on QoE must be considered as well. However, almost all the developed QoE monitoring tools in the literature [15–19] did not take the effects of AMP on QoE into account. Thus, in [23] we proposed the QoE monitoring system that takes the effects of AMP on QoE into consideration. Additionally, the most important feature of our proposed QoE monitoring system is that it can derive an overall QoE for several QoS metrics using the product form presented in [5]. Our previous work [5] has shown that the product form performs much better than the conventional averaging techniques in deriving an overall QoE of multiple QoS metrics. Using the developed QoE monitoring program, network providers can identify network troubles and fix them in real time to improve the system performance and qualities.

The novelties and contributions of this paper are summarized as follows. First, the implemented QoE monitoring system takes the effects of AMP on QoE into consideration. Second, the developed QoE monitoring system derives an overall QoE from several QoS metrics such as the initial playout delay, packet loss rate, underflow time ratio, and normalized playout rate. Thirdly, we implement the AMP scheme, which was proposed in [22], into the VLC media player. Finally, the Group of Pictures- (GOP-) based cumulative average jitter J_n, which is used to determine the playback threshold P_n of the proposed AMP, is proposed to eliminate the effect of diverse frame sizes on the estimation of network jitter.

The rest of this paper is organized as follows. Section 2 describes the implementation of the AMP playout system. Implementation of the QoE monitoring system is then introduced in Section 3. Section 4 conducts several experiments and evaluates the QoE performance of video streaming services using the developed programs. Finally, the concluding remarks are given in Section 5.

2. Implementation of AMP

This section first describes the principles of AMP in Section 2.1. Then the implementation schemes, available

programming tools, and some challenging issues on measurements or estimations of key parameters are addressed in Section 2.2.

2.1. Principles of Proposed AMP. In this work, we employ the freeware VLC media player to realize the AMP. The AMP is implemented according to our proposed scheme [22] whose principle is described briefly as follows. In addition to the conventional slowdown threshold L, a dynamic playback threshold P_n and a speedup threshold H are designed. If the buffer fullness is below L, the playout rate is reduced to avoid buffer underflow. When the buffer fullness exceeds the speedup threshold H, the playout rate is increased to reduce the buffer fullness for avoiding buffer overflow. The Dynamic Playback Threshold Algorithm (DPTA) in [22] is designed to dynamically adjust the playback threshold P_n, which affects the initial playout delay, under various network conditions. Whenever the current buffer fullness n surpasses or equals P_n, the playback starts immediately with the proper playout rate $\mu(i, n)$ which is determined by the following equation:

$$\mu(i, n) = \begin{cases} R_L(n) & \text{if } n < L, \\ R_S(i) & \text{if } L \le n \le H, \\ R_H(n) & \text{if } n > H, \end{cases} \tag{1}$$

where $R_S(i)$ is a random process and is estimated by the Arrival Process Tracking Algorithm (APTA) presented in [22]. That is, $R_S(i)$ depends on the frame arrival process at the client buffer and is limited between $(1 - r)\mu_0$ and $(1 + r)\mu_0$, as shown in Figure 1. Anyone interested in the detailed derivation of $R_S(i)$ and P_n can refer to [22]. The quadratic playout rate functions $R_H(n)$ and $R_L(n)$ in [22] are defined as follows:

$$R_H(n) = (1 + r_2)\mu_0 - \left(\frac{N - \min\{n, N\}}{N - H}\right)^2 (r_2 - r)\mu_0,$$
$$R_L(n) = (1 - r_1)\mu_0 + \left(\frac{n}{L}\right)^2 (r_1 - r)\mu_0, \tag{2}$$

where N is the uppermost threshold correlated to the buffer size at the client. Notably, the playout rate must be limited such that rate variations are unnoticeable or acceptable by users. The perception of a slowdown video is usually different from that of a speedup video for users. Therefore, two different restricted deviation ratios, denoted by r_1 and r_2 for $R_L(n)$ and $R_H(n)$, respectively, are set for playout rates. The corresponding playout rate function of the proposed AMP is given by Figure 1. Numerical results in [22] have shown that the proposed AMP with quadratic playout rates performs much better than several conventional playout systems with linear playout rates and can adapt to different network load conditions for reducing the initial playout delay.

2.2. Realization of AMP. VLC media player provides a useful VLC ActiveX package whose architecture is given in Figure 2. The main component of VLC ActiveX package is LibVLC which supports several properties such as *input,*
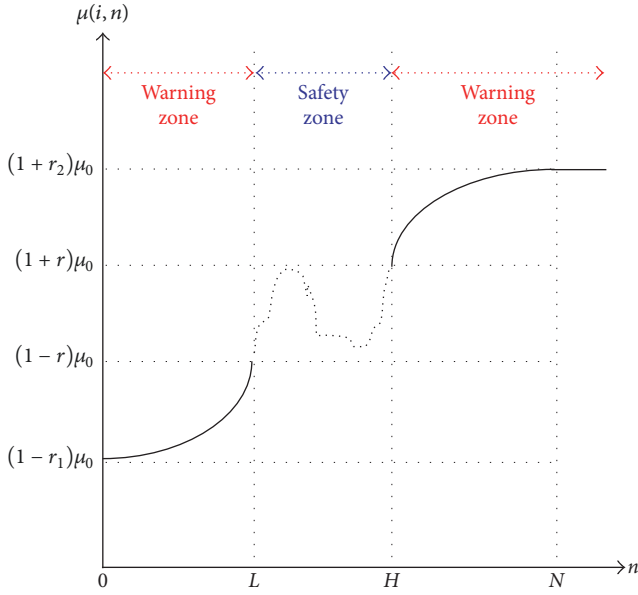
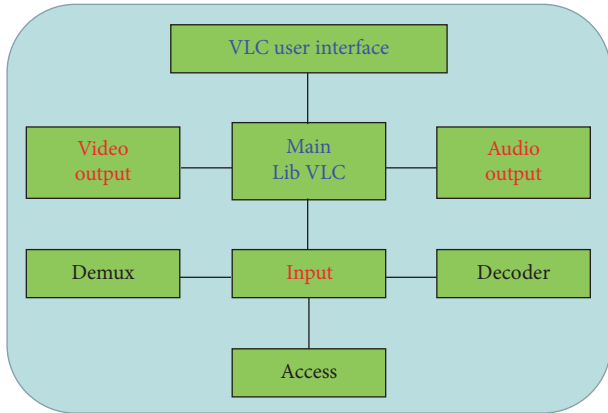FIGURE 1: Playout rate function of the proposed AMP [22].

FIGURE 2: Architecture of VLC ActiveX package.

TABLE 1: Receiving parameters in VLC *input.item():stats()*.

Parameters	Meaning
demux_read_packets	Number of packets received
demux_read_bytes	Number of bytes received
demux_bitrate	Bit rate of the video stream
average_demux_bitrate	Average bit rate of the video stream
demux_corrupted	Number of corrupted frames
decoded_audio	Number of audio frames decoded
decoded_video	Number of video frames decoded
lost_pictures	Number of frames lost

a new MPEG-2 TS video frame is arriving. Hence, the buffer fullness n can be computed by the following equation:

$$n = demux\_frames - decoded\_video. \tag{3}$$

After deriving the buffer fullness n, the playout rate of the AMP media player can be dynamically adjusted according to (1). The command to dynamically set the playout rate of the VLC media player is defined by the function *var.set(input, "rate", $\mu(i,n)$)*. Although the currently developed program is designed for the MPEG-2 TS format, there is no difficulty to include other advanced video encoding technologies such as the H.264/AVC and H.265/HEVC [24] in the future.

The dynamic playback threshold P_n [22] is determined by

$$P_n = L_0 + (L - L_0) \cdot \min\left\{1, \frac{J_n}{cT}\right\}, \tag{4}$$

where L_0 and c are design parameters and $T = 1/\mu_0$ is the average frame interarrival time. J_n is the cumulative average jitter and is defined by

$$J_n = \left(1 - \frac{1}{n-1}\right)J_{n-1} + \frac{|X_{n-1} - T|}{n-1}, \quad \text{for } n > 1, \tag{5}$$

where X_{n-1} is the frame interarrival time between the $(n-1)$-st and the nth frames. However, video frames are categorized into I, B, and P frames and different types of frames usually have very different frame sizes. For example, the frame size of an I frame is usually much larger than that of a B or P frame. Hence, the frame interarrival time between two successive I and B (or P) frames may become relatively large because of the long transmission time of an I frame, compared with that between two successive B and P frames. Such variance of frame interarrival times maybe is not due to the network jitter but the diversity of frame sizes. To reduce the impact of frame size variation on the estimation of network jitter J_n and estimate J_n more accurately, in this work the average frame interarrival time within a GOP is used. Hence, the variable X_{n-1} in (5) is replaced by Y_{k-1}/G, where $k = \lfloor n/G \rfloor$, G is the number of frames in a GOP, and Y_{k-1} is the interarrival time between the I frames of the $(k-1)$-st and the kth GOPs.

3. Implementation of QoE Monitoring System

The procedures to implement the QoE monitoring system are described as follows. First, we propose an architecture
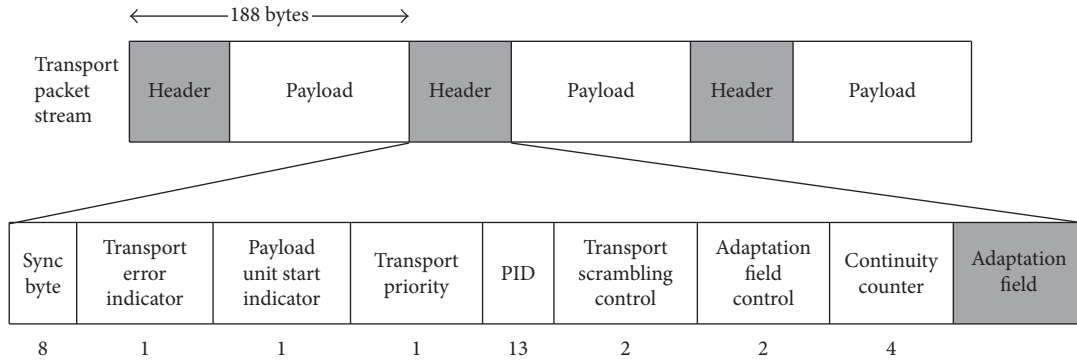
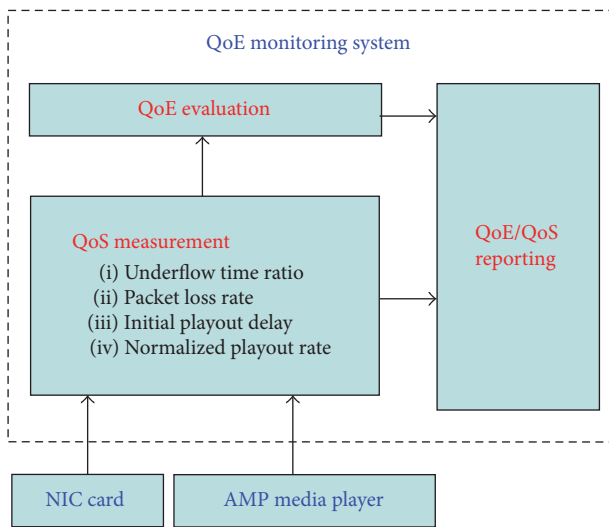*audio output,* and *video output.* The *input* property collects all related parameters of the received video stream from the Network Interface Card (NIC) *(Access),* the network demultiplexer *(Demux),* and the video decoder *(Decoder),* as shown in Figure 2. Several useful statistical parameters in *input.item():stats()* are listed in Table 1. Based on these collected parameters from the *input* property, one can set and control the playout rate of the VLC media player to realize the AMP function.

According to the descriptions in Section 2.1, the playout rate mainly depends on the buffer fullness n (in frames). To accurately measure the number *demux_frames* of demultiplexed video frames, one can directly analyze the packets arriving at the NIC card. By reading the "payload unit start indicator" bit in the header of each 188-byte MPEG-2 TS (Transport Stream) packet, as shown in Figure 3, one can easily calculate the number *demux_frames* of video frames received. When the "payload unit start indicator" bit equals 1,

FIGURE 3: MPEG-2 TS packet format.

FIGURE 4: Architecture of the QoE monitoring system.

FIGURE 5: Buffering mechanism in VLC player.

for the QoE monitoring system. Next, several network-based QoS metrics are defined and their measurement methods are presented. Finally, the overall QoE evaluation method for the QoE monitoring system is introduced.

3.1. Architecture of QoE Monitoring System.

The architecture of our proposed QoE monitoring system is plotted in Figure 4. It includes the QoS measurement engine, QoE evaluation function, and performance reporting window. The QoS measurement engine is designed to collect the QoS metrics from the media player and the NIC card. These obtained QoS metrics are used by the QoE evaluation function for producing the overall QoE of video streaming services. All instantaneous QoS metrics and QoE results are reported in real time and the curve of overall QoE versus time is also displayed with a chart in the Graphic User Interface (GUI) for the monitoring purpose.

3.2. Measurements of QoS Parameters.

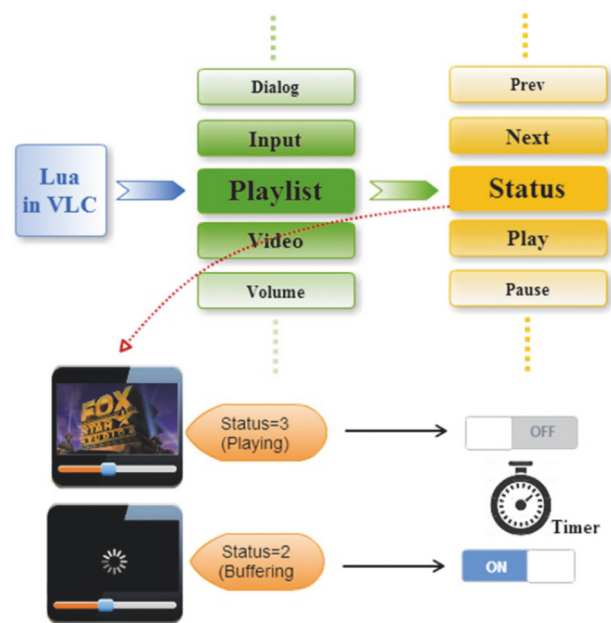In the proposed QoE monitoring system, the considered QoS metrics include underflow time ratio QoS_u, packet loss rate QoS_l, initial playout delay QoS_d, and normalized playout rate QoS_r. The measurements of these parameters are described as follows.

(a) Underflow time ratio QoS_u: whenever an underflow event occurs, the underflow time starts to accumulate until the playback restarts again. At the same time, the number of underflow events is increased by 1. The underflow time ratio equals the ratio of accumulated underflow time to the total playback time duration. The accumulated underflow time can be computed according to the value of *Status*() provided by VLC Lua extension program, as shown in Figure 5. When the value of *Status*() changes into 2, the underflow timer turns ON until the value of *Status*() becomes 3. The sojourn time at the state with *Status*() = 2 is the underflow time. Whenever the underflow timer turns OFF, the accumulated underflow time is updated.

(b) Packet loss rate QoS_l: the packet loss rate is obtained by analyzing the number of receiving packets at the

NIC card or using the network protocol analyzer such as Wireshark.

(c) Initial playout delay QoS_d: initial playout delay strongly depends on the buffering time set in VLC media player. However, in our implemented AMP this parameter is directly determined by the DPTA algorithm proposed in [22].

(d) Normalized playout rate QoS_r: whenever the playout rate is adjusted, the normalized playout rate is computed and saved. However, within each minute only the maximum and minimum normalized playout rates are used in our design [5]. That is, the corresponding QoE of QoS_r during every minute equals the mean of these two individual QoEs resulting from the maximum and minimum normalized playout rates.

3.3. Overall QoE Evaluation Function. The overall QoE evaluation function adopts the integrated multivariate QoE function $f_I(QoS_u, QoS_l, QoS_d, QoS_r)$ and the individual QoS-QoE mapping functions $f_u(x)$, $f_l(x)$, $f_d(x)$, and $f_r(x)$ proposed in [5] to derive the overall QoE. Based on the results of [5], they are given as follows:

$$f_I \left(QoS_u, QoS_l, QoS_d, QoS_r \right)$$

$$= 5 \cdot \frac{f_u \left(QoS_u \right)}{5} \cdot \frac{f_l \left(QoS_l \right)}{5} \cdot \frac{f_d \left(QoS_d \right)}{5} \quad (6)$$

$$\cdot \frac{f_r \left(QoS_r \right)}{5},$$

$$f_u(x) = 5e^{-5.71x}, \quad (7)$$

$$f_l(x) = 5e^{-1.607x}, \quad (8)$$

$$f_d(x) = 5e^{-0.0416x}, \quad (9)$$

$$f_r(x) = 5x^{8.94}e^{-8.94(x-1)}. \quad (10)$$

The scheme used in [5] consists of three steps to be done beforehand. First, several Key Performance Indicators (KPIs) or QoS metrics, such as the initial playout delay, packet loss rate, underflow time ratio, and playout rate, were selected. Then several distorted videos were generated based on these selected QoS metrics. Second, subjective tests were conducted using human observers to rate the distorted videos in terms of Mean Opinion Score (MOS) [25]. The ACR subjective test method was adopted in [5]. However, the length of each distorted video clip is about 64 s rather than 10 s. Each distorted video clip was viewed and rated by no less than 30 viewers. The title of the used video clip, which is with the frame rate 30 fps, resolution 1920 × 1080 p, and video codec MPEG-2, is "Avatar." Thirdly, using the regression approach, the QoS-QoE mapping functions for individual QoS metrics were obtained. Finally, the product form of individual QoS-QoE mapping functions [5, 6] is used to derive the overall QoE of a video streaming service. Notably, although only four QoS metrics are considered in
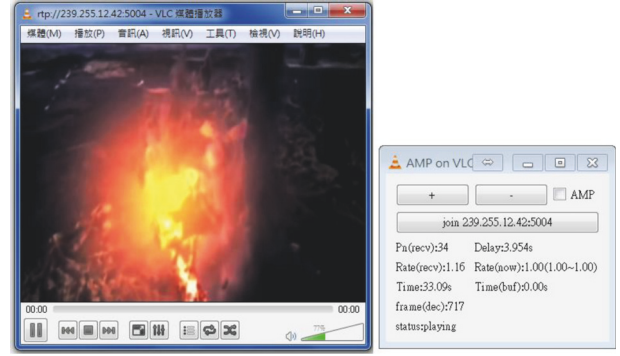

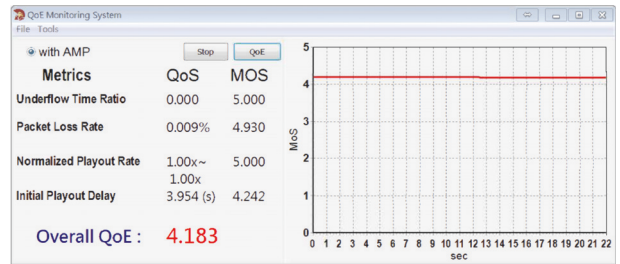FIGURE 6: VLC media player and Lua program.

FIGURE 7: QoE/MOS monitoring program.

(6), the number of QoS metrics can be arbitrarily extended as needed.

4. Experimental Results

In this section, we first briefly introduce the GUIs and functions of developed AMP and QoE monitoring programs. Subsequently, using the developed QoE monitoring program, several experiments are conducted to demonstrate the performance of AMP. Finally, we use the QoE monitoring program to identify some challenging issues of wireless streaming.

4.1. QoE Monitoring System. The developed AMP is based on the VLC media player and its add-ons such as the Lua extension. Figure 6 shows the VLC media player and its Lua extension program. One can directly set the multicast IP of the streaming source and activate the AMP function on the VLC Lua extension program. Although in this paper the AMP is implemented into the VLC media player, there is no difficulty to implement it into other popular media players such as the MS Windows Media Player or YouTube Player. The QoE monitoring program, which is developed using the C++ Builder and is shown in Figure 7, evaluates and displays the overall QoE. The measured QoS and synthetic QoE/MOS are displayed in real time on the monitoring window, as shown in Figure 7. The corresponding QoEs of individual QoSs are displayed on the left side of the monitoring window while the curve of the overall QoE versus time is displayed on the right side of the monitoring window, as shown in Figure 7. By clicking the "Tools" on the menu bar of the QoE monitoring program, other detailed QoS metrics
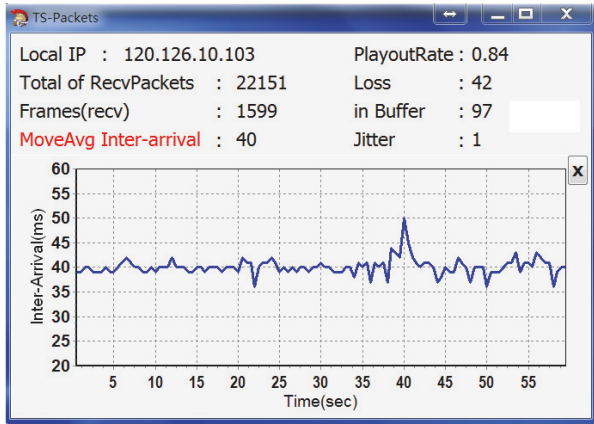
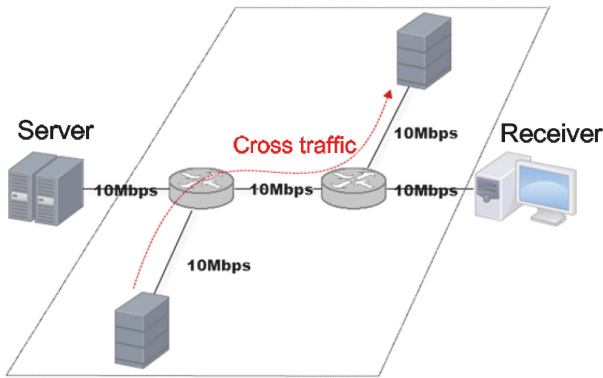FIGURE 8: Monitoring tools for video packets at the client.

FIGURE 9: Experimental network scenario 1.

TABLE 2: Information of test video 1.

Parameter	Value
Video title	The Big Bang Theory
Length	593 sec
Resolution	352×288 pixels
Encoding scheme	MPEG-2 TS
Bit rate	1406 kbps
Frame rate	25 fps
GOP size	15
Content size	99.5 MB

such as the interarrival time of video frames, normalized playout rate, number of packet losses, buffer fullness, and the jitter observed at the client can be displayed, as shown in Figure 8. By clicking the interested QoS parameter, such as the "MoveAvg Interarrival," the curve of the QoS versus time is immediately displayed on the monitoring window of Figure 8.

4.2. Performance Evaluation of AMP. In order to demonstrate our AMP design and QoE monitoring system, the first experimental network scenario is constructed, as shown in

Figure 9. The server pumps the video stream into the first router, followed by a bottleneck link and the second router. Finally, the disturbed video stream arrives at the receiver. All link capacities in Figure 9 are set to be 10 Mbps. The Pareto ON-OFF cross-traffic is injected into the bottleneck link using the Distributed Internet Traffic Generator (D-ITG) tool [26]. Therefore, the video stream is disturbed by the cross-traffic in the bottleneck link and thus packet losses and jitters may occur. The video is encoded into the MPEG-2 TS at the streaming server and the RTP/UDP streaming protocol is employed. Table 2 lists the detailed information of the first test video and Table 3 lists the parameters of AMP. The other parameters not mentioned here are set the same as those in [22].

First, the QoEs of two media playout systems with and without AMP are compared. The video server pumps a multicast stream into the network and two independent clients simultaneously receive the same multicast stream at the remote side. One client uses the media player with AMP, while the other one adopts the media player without AMP. The observed QoEs of these two systems with and without AMP under different cross-traffic rates 0, 4, and 8 Mbps are given in Figure 10. When the cross-traffic rate is not larger than 4 Mbps, the overall QoE is almost over 4, no matter whether the AMP is enabled or not. This is because the jitter, burstiness, and losses of video packets are very small under a low cross-traffic rate. Additionally, under a low cross-traffic rate the observed QoE on the media player without AMP remains very stable while the QoE on the media player with AMP slightly oscillates in the first minute. The reason is explained as follows. The media player without AMP can start playback only when the buffer fullness exceeds $L = 100$, yielding the initial playout delay about 4 s, and always plays the video with the normal playout rate. However, the media player with AMP can start playback earlier but with a lower initial playout rate. Since the initial playout rate of the media player with AMP is smaller than the normal playout rate, the observed QoE on the media player with AMP is slightly degraded in the first minute, as shown in Figure 10. Table 4 shows the measured initial playout delays QoS_d of the media players with and without AMP. Because the initial playout delay in the media player with AMP is much smaller than that without AMP, the corresponding QoE of QoS_d of the media player with AMP becomes better. When the cross-traffic rate becomes 8 Mbps, the video stream is seriously disturbed by the cross-traffic so that the observed QoE at the client becomes unacceptable (MOS < 3). The serious QoE degradation is mainly due to the increase of the number of underflow events and underflow time. However, the observed QoE on the media player with AMP is still much better than that without AMP, as shown in Figure 10.

The corresponding buffer fullness and the playout rate at the client under the cross-traffic rate 6 Mbps are shown in Figure 11. According to Figure 11, the buffer fullness of the system with AMP is more stable than that of the system without AMP. For the media player with AMP, the underflow events never happen because the buffer fullness never drops to an unacceptable low level after the playback. Although the playout rate of the media player with AMP varies, the
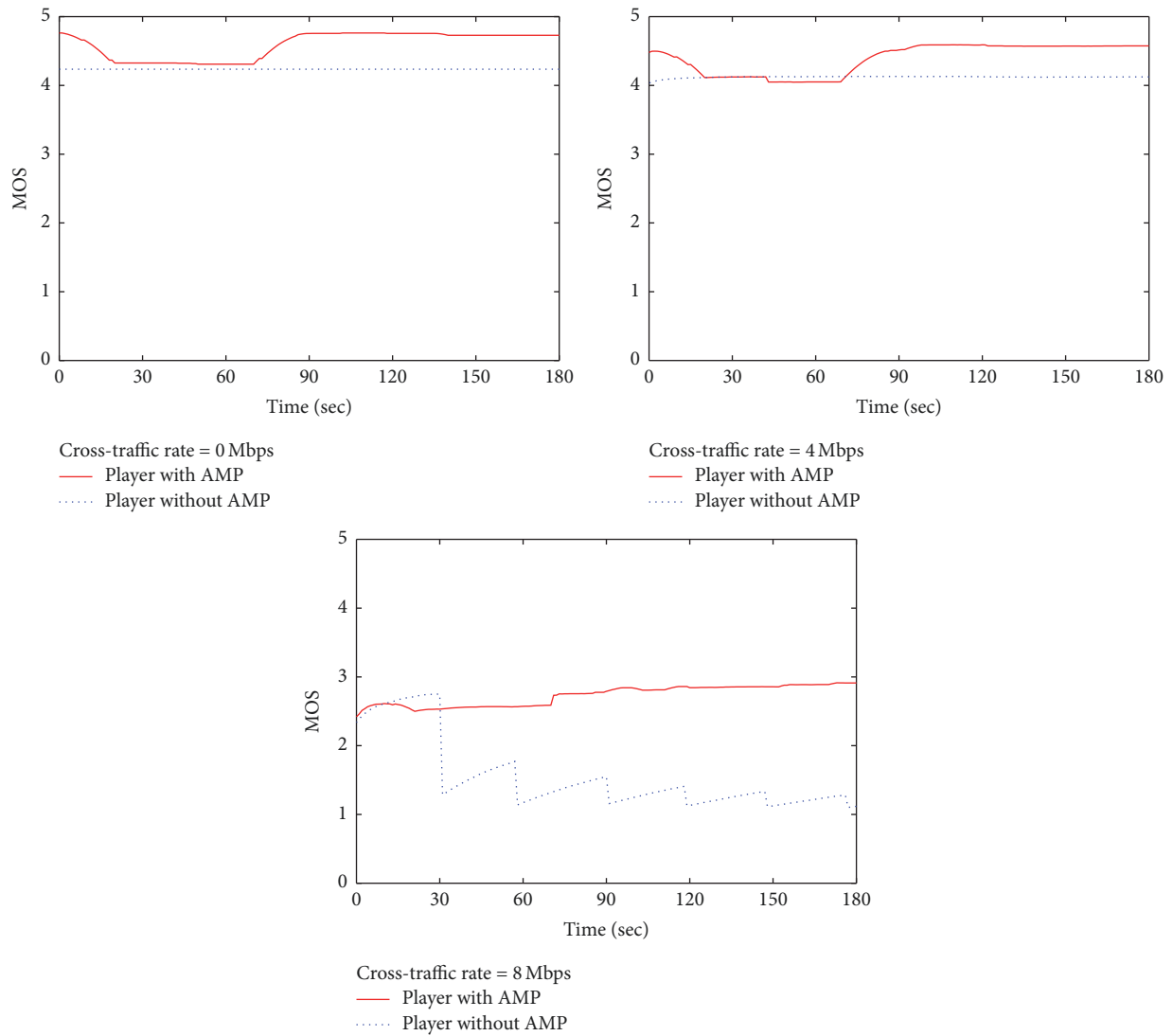
Cross-traffic rate = 0 Mbps
—— Player with AMP
······ Player without AMP

Cross-traffic rate = 4 Mbps
—— Player with AMP
······ Player without AMP

Cross-traffic rate = 8 Mbps
—— Player with AMP
······ Player without AMP

FIGURE 10: Comparison of synthetic MOSs/QoEs between the systems with and without AMP under various cross-traffic rates.

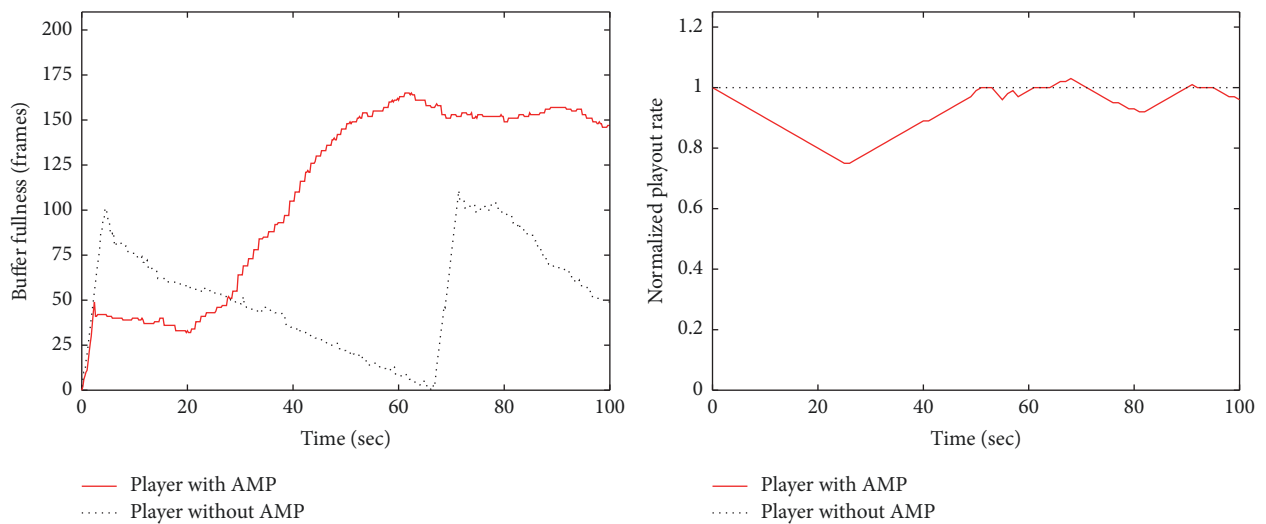—— Player with AMP
······ Player without AMP

—— Player with AMP
······ Player without AMP

FIGURE 11: Comparison of buffer fullness and normalized playout rates between systems with and without AMP (cross-traffic rate = 6 Mbps).

Table 3: Parameters setting in AMP.

Parameter	L_0	L	H	N	c	r	r_1	r_2
Value	30	100	200	500	0.4	0.15	0.3	0.3

Table 4: Initial playout delay QoS_d and corresponding QoE.

Cross-traffic rate (Mbps)	Player with AMP QoS_d/QoE	Player without AMP QoS_d/QoE
0	1.124 sec/4.776	4.001 sec/4.233
4	1.656 sec/4.667	4.250 sec/4.189
8	3.876 sec/4.255	4.756 sec/4.102

Table 5: Information of test video 2.

Parameter	Value
Video title	Call Me Maybe (MV)
Length	199 sec
Resolution	720×480 pixels
Encoding scheme	MPEG-2 TS
Bit rate	1936 kbps
Frame rate	30 fps
GOP size	15
Content size	46 MB

deviation of the playout rate is never over 25% so that the QoE degradation resulting from it is still acceptable. However, for the media player without AMP, the buffer fullness during the time interval (10 s, 66 s) decreases almost linearly to zero and leads to an underflow event, yielding a significant QoE degradation. All results in Figures 10 and 11 demonstrate that the AMP significantly improves the QoE of streaming services.

4.3. Performance Monitoring for Wireless Streaming. Video streaming over wireless networks has become very popular recently. However, it is still challenging to offer streaming services over wireless networks. To clearly identify the challenging issues of wireless streaming services, in this subsection the AMP function is not enabled in the media players of all receivers. To evaluate the QoE of wireless streaming services, the second network scenario shown in Figure 12 is considered and the first test video given in Table 2 is used. The video stream is multicast to two independent receivers at the last hop: via the IEEE 802.11n wireless link to Receiver 1 and via the wired link to Receiver 2. Receiver 1 is away from the WiFi AP by 2 m and no obstacle between them. Figure 13 shows the observed synthetic MOSs at Receivers 1 and 2 under different cross-traffic rates, using the QoE monitor program developed in this paper. By Figure 13, the QoE at Receiver 2 using a wired link is better than that at Receiver 1 using a wireless link. Even though no cross-traffic exists, the MOS of the video streaming service at Receiver 1 via the wireless link is less than 4. According to the QoS parameters measured at Receivers 1 and 2, the difference of QoE between them is mainly due to a higher packet loss rate in the wireless link.

Subsequently, to compare the performance of multicast and unicast, the network scenario 3 shown in Figure 14 is considered. The second video clip with the resolution 720 × 480 p and bit rate 1936 kbps is used, as shown in Table 5. Different numbers of similar unicast streams are pumped from the video server to the receivers using a wireless link at the last hop. The synthetic MOS of the streaming service at the receivers decreases significantly as the number of unicast streams increases, as shown in Figure 15. This is because the jitter, burstiness, and packet loss rate increase as the number of unicast streams in the network grows, yielding the QoE degradation at the receivers. Note that if the multicast scheme

is used, only one video stream is required to be pumped into the network no matter how many receivers exist. That is, the QoE of video streaming services is irrelevant to the number of receivers if the multicast scheme is employed. Obviously, under the scenario that multiple users watch the same video stream, the multicast scheme performs much better than the unicast one in terms of QoE.

Finally, to investigate the impact of transmission distance and obstacles on the QoE of wireless streaming services, the wireless network scenario 4 plotted in Figure 16 is considered and the second test video given in Table 5 is used. The video stream is multicast to several wireless receivers at different positions in our labs. No cross-traffic is injected into the network. No obstacles exist between the wireless AP and the receivers at Locations 1 and 2. However, the receiver at Location 1 is much closer to the AP than the receiver at Location 2 is. There are cupboards and a wall between the AP and the receiver at Location 3. The receiver at Location 4 is outside the door of our lab and is isolated from the AP by a wall. The other receiver at Location 5 is also isolated from the AP by a wall and cupboards. All receivers receive the same multicast stream simultaneously and their individual MOSs are estimated using the developed QoE monitoring program. The results are shown in Figure 17, where the MOSs of the receivers located at Locations 1 and 2 are the best because no obstacle blocks the wireless signals for them, while the MOSs of the receivers located at Locations 4 and 5 are the worst because there exist cupboards and a wall between them and the AP. The MOS of the receiver located at Location 3 is better than those located at Locations 4 and 5 because the receiver at Location 3 is much closer to the AP than those at Locations 4 and 5 are. Accordingly, the distance and the number of obstacles between the WiFi AP and the receivers strongly affect the QoE of wireless streaming services.

5. Conclusions

In this work, an AMP and a real-time QoE monitoring system for video streaming services arerealized. The AMP design is
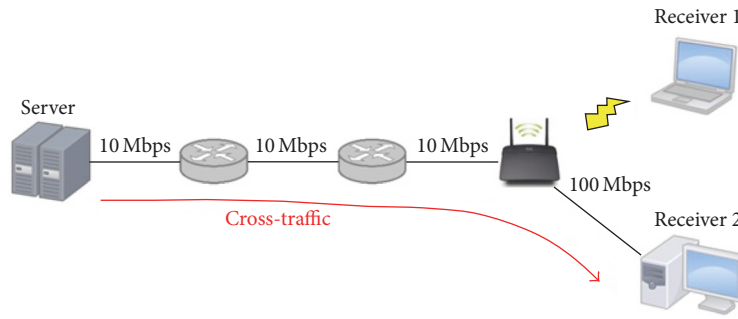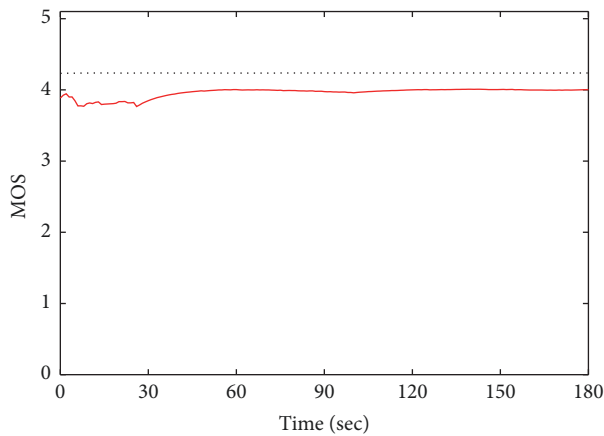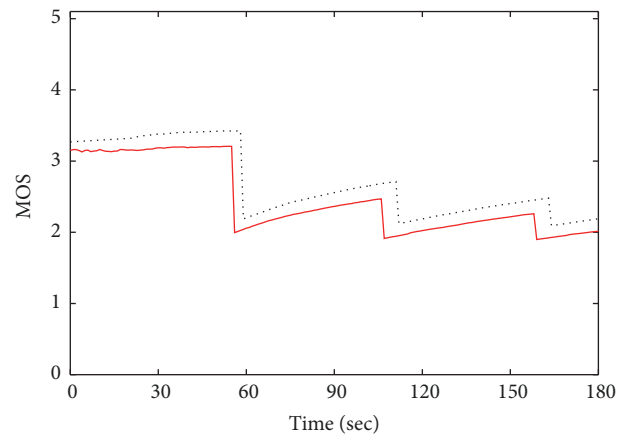
FIGURE 12: Experimental network scenario 2.

Cross-traffic rate = 0 Mbps
..... Receiver 2 (wired)
⎯⎯ Receiver 1 (wireless)

Cross-traffic rate = 6 Mbps
..... Receiver 2 (wired)
⎯⎯ Receiver 1 (wireless)

FIGURE 13: Comparison of QoEs of streaming services between wired and wireless receivers under different cross-traffic rates.
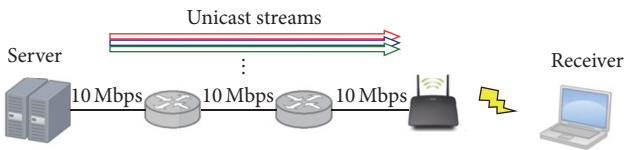

FIGURE 14: Experimental network scenario 3.

−·− 1 unicast stream   ⎯⎯ 5 unicast streams
..... 3 unicast stream

FIGURE 15: Impact of number of unicast streams on MOS of wireless receivers.

implemented into the freeware VLC media player. The QoE monitoring system can measure several key QoS metrics such as the packet loss rate, underflow time ratio, initial playout delay, and the playout rates of a video streaming service and derive the overall QoE in real time. Several experiments are conducted in various scenarios including wired and wireless networks. All numerical results demonstrate that the QoE monitoring system works well and the proposed AMP significantly improves the QoE of video streaming services. Using the real-time QoE monitoring system, network/content providers can find the network troubles and resolve them timely to maintain good QoEs for users. In the future, more advanced video codecs such as H.264/AVC and H.265/HEVC will be implemented into the developed system.
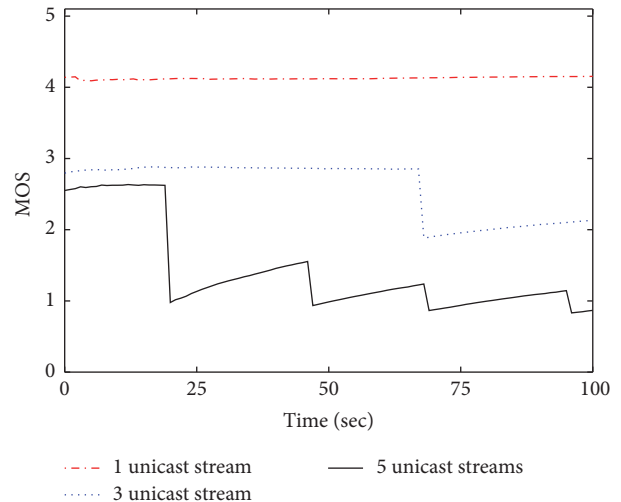
Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.
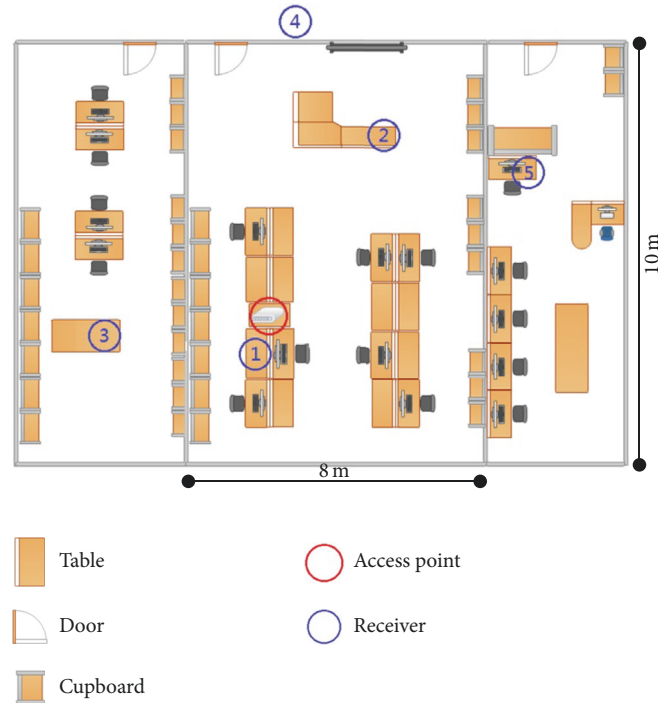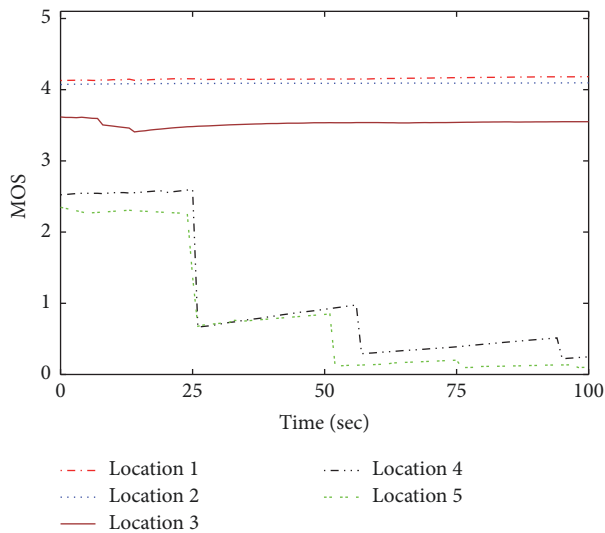
FIGURE 16: Experimental network scenario 4.

FIGURE 17: Impact of transmission distances and obstacles on QoE of wireless streaming services.
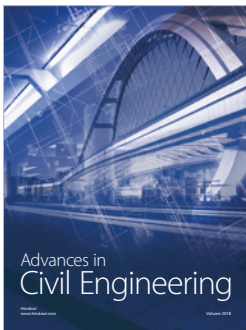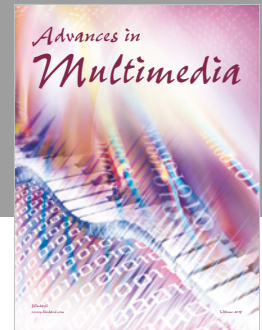
Acknowledgments

References

[1] Y. Xiao, X. Du, J. Zhang, F. Hu, and S. Guizani, "Internet protocol television (IPTV): the killer application for the next-generation internet," *IEEE Communications Magazine*, vol. 45, no. 11, pp. 126–134, 2007.

[2] M. Li and C.-H. Wu, "A cost-effective resource allocation and management scheme for content networks supporting IPTV services," *Computer Communications*, vol. 33, no. 1, pp. 83–91, 2010.

[3] S. Park and S.-H. Jeong, "Mobile IPTV: Approaches, challenges, standards, and QoS support," *IEEE Internet Computing*, vol. 13, no. 3, pp. 23–31, 2009.

[4] G. Gardikis, G. Xilouris, E. Pallis, and A. Kourtis, "Joint assessment of Network- and Perceived-QoS in video delivery networks," *Telecommunication Systems*, vol. 49, no. 1, pp. 75–84, 2012.

[5] M. Li and C.-Y. Lee, "A cost-effective and real-time QoE evaluation method for multimedia streaming services," *Telecommunication Systems*, vol. 59, no. 3, pp. 317–327, 2015.

[6] M. Li, "QoE-based performance evaluation for adaptive media playout systems," *Advances in Multimedia*, vol. 2013, Article ID 152359, 2013.

[7] M. Mu, J. Ishmael, W. Knowles et al., "P2P-based IPTV services: Design, deployment, and QoE measurement," *IEEE Transactions on Multimedia*, vol. 14, no. 6, pp. 1515–1527, 2012.

[8] W.-H. Hsu and C.-H. Lo, "QoS/QoE mapping and adjustment model in the cloud-based multimedia infrastructure," *IEEE Systems Journal*, vol. 8, no. 1, pp. 247–255, 2014.

[9] Y. Ju, Z. Lu, D. Ling, X. Wen, W. Zheng, and W. Ma, "QoE-based cross-layer design for video applications over LTE," *Multimedia Tools and Applications*, vol. 72, no. 2, pp. 1093–1113, 2014.

[10] J. Lloret, A. Canovas, J. Tomas, and M. Atenas, "A network management algorithm and protocol for improving QoE in mobile IPTV," *Computer Communications*, vol. 35, no. 15, pp. 1855–1870, 2012.

[11] M. A. Hoque, M. Siekkinen, J. K. Nurminen, M. Aalto, and S. Tarkoma, "Mobile multimedia streaming techniques: QoE and energy saving perspective," *Pervasive and Mobile Computing*, vol. 16, pp. 96–114, 2015.

[12] L. Qian, H. Chen, and L. Xie, "SVM-based QoE estimation model for video streaming service over wireless networks," in *Proceedings of the IEEE International Conference on Wireless Communications Signal Processing (WCSP)*, Nanjing, China, 2015.

[13] Y. Chen, K. Wu, and Q. Zhang, "From QoS to QoE: A tutorial on video quality assessment," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 1126–1165, 2015.

[14] ITU-T Rec. P.910, "Subjective video quality assessment methods for multimedia applications," 2008.

[15] D. Saladino, A. Paganelli, and M. Casoni, "A tool for multimedia quality assessment in NS3: QoE Monitor," *Simulation Modelling Practice and Theory*, vol. 32, pp. 30–41, 2013.

[16] M. Schmitt, S. Gunkel, P. Cesar, and P. Hughes, "A QoE testbed for socially-aware video-mediated group communication," in *Proceedings of the 2nd International Workshop on Socially-Aware Multimedia, SAM 2013 - Co-located with ACM Multimedia 2013*, pp. 37–41, Spain, October 2013.

[17] S. Jeong and H. Ahn, "Mobile IPTV QoS/QoE monitoring system based on OMA DM protocol," in *Proceedings of the 2010 International Conference on Information and Communication Technology Convergence, ICTC 2010*, pp. 99-100, Republic of Korea, November 2010.

[18] M. Leszczuk, M. Hanusiak, M. C. Q. Farias, E. Wyckens, and G. Heston, "Recent developments in visual quality monitoring by key performance indicators," *Multimedia Tools and Applications*, vol. 75, no. 17, pp. 10745–10767, 2016.

[19] S. Ickin, M. Fiedler, K. Wac, P. Arlos, C. Temiz, and K. Mkocha, "VLQoE: Video QoE instrumentation on the smartphone," *Multimedia Tools and Applications*, vol. 74, no. 2, pp. 381–411, 2014.

[20] VideoLAN- VLC medial player official site, http://www.videolan.org/.

[21] J. Yang, H. Hu, H. Xi, and L. Hanzo, "Online buffer fullness estimation aided adaptive media playout for video streaming," *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 1141–1153, 2011.

[22] M. Li, T.-W. Lin, and S.-H. Cheng, "Arrival process-controlled adaptive media playout with multiple thresholds for video streaming," *Multimedia Systems*, vol. 18, no. 5, pp. 391–407, 2012.

[23] M. Li, C.-L. Yeh, and S.-Y. Lu, "Realization of intelligent media player and QoE monitoring system for multimedia streaming services," *IEEE ICCP & HSIC*, pp. 1–4, 2013.

[24] P. Seeling and M. Reisslein, "Video traffic characteristics of modern encoding standards: H.264/AVC with SVC and MVC extensions and H.265/HEVC," *The Scientific World Journal*, vol. 2014, Article ID 189481, 16 pages, 2014.

[25] ITU-T Rec. P.800, "Mean opinion score (MOS) terminology," 2003.

[26] A. Botta, A. Dainotti, and A. Pescapé, "A tool for the generation of realistic network workload for emerging networking scenarios," *Computer Networks*, vol. 56, no. 15, pp. 3531–3547, 2012.