Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Seamless handover between unicast and multicast multimedia streams*

Mau-Luen THAM[1], Chee-Onn CHOW[1], Yi-han XU[2], Khong Neng CHOONG[3], Cheng Suan LEE[4]

([1]*Department of Electrical Engineering, Faculty of Engineering, University of Malaya, Kuala Lumpur 50603, Malaysia*)

([2]*College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China*)

([3]*Wireless Communication Cluster, MIMOS Berhad, Kuala Lumpur 57000, Malaysia*)

([4]*Power-All Networks, Jalan Ampang, Kuala Lumpur 50450, Malaysia*)

E-mail: thammauluen@siswa.um.edu.my; cochow@um.edu.my; xuyihan86@gmail.com;
kn.choong@mimos.my; cslee@powerallnetworks.com

Received Feb. 16, 2014; Revision accepted May 24, 2014; Crosschecked Sept. 17, 2014

Abstract: With the deployment of heterogeneous networks, mobile users are expecting ubiquitous connectivity when using applications. For bandwidth-intensive applications such as Internet Protocol Television (IPTV), multimedia contents are typically transmitted using a multicast delivery method due to its bandwidth efficiency. However, not all networks support multicasting. Multicasting alone could lead to service disruption when the users move from a multicast-capable network to a non-multicast network. In this paper, we propose a handover scheme called application layer seamless switching (ALSS) to provide smooth real-time multimedia delivery across unicast and multicast networks. ALSS adopts a soft handover to achieve seamless playback during the handover period. A real-time streaming testbed is implemented to investigate the overall handover performance, especially the overlapping period where both network interfaces are receiving audio and video packets. Both the quality of service (QoS) and objective-mapped quality of experience (QoE) metrics are measured. Experimental results show that the overlapping period takes a minimum of 56 and 4 ms for multicast-to-unicast (M2U) and unicast-to-multicast (U2M) handover, respectively. The measured peak signal-to-noise ratio (PSNR) confirms that the frame-by-frame quality of the streamed video during the handover is at least 33 dB, which is categorized as good based on ITU-T recommendations. The estimated mean opinion score (MOS) in terms of video playback smoothness is also at a satisfactory level.

Key words: Experimental approach, Multimedia session continuity, Seamless handover, Unicast/multicast switching, Multimedia streaming
doi:10.1631/jzus.C1400052 **Document code:** A **CLC number:** TN929.5

1 Introduction

Mobile terminals (MTs) are rapidly evolving toward supporting multimode operations with the adoption of multiple air interface technologies within a single mobile device. In combination with the omnipresence of heterogeneous access networks such as Wi-Fi, WiMax, and 3G, users can access multimedia services anywhere at any time through any network. Internet Protocol Television (IPTV) is one of the popular applications, in which multimedia content is transmitted to its subscribers via either unicast or multicast delivery methods.

Multicast seems to be the best way to deliver multimedia services to a large number of users due to its bandwidth efficiency (Hilt *et al.*, 2009). While multimedia content delivery could gain performance benefits with multicast, such capability is not consistently available across the entire network infrastructure (Namburi *et al.*, 2006). For example, although 3G networks support Multimedia Broadcast Multicast Services (MBMS) for Mobile TV (MTV)

* Project supported by the Ministry of Science, Technology and Innovation under the eScienceFund (No. 01-01-03-SF0782) and MIMOS Berhad

services, they are not activated in certain geographical areas or cells (Pitsillides and Christophorou, 2007). This is owing to the costing issue that causes a network operator to provide MBMS only in areas with high subscriber density (Åström and Edlund, 2009). This limitation could lead to service disruption when a user moves from a multicast-enabled network to a non-multicast network. Hence, a handover scheme that takes into account unicast and multicast delivery methods is mandatory.

One of the key issues in handover is to maintain the ongoing communication quality in order to make it transparent to the connected end-users. This is strongly related to the types of handover being invoked, which can be classified into hard handover and soft handover. A hard handover permits only connection to one active link at a time. In other words, the connection from the old link must be released before the new one is established (thus such handovers are also termed break-before-make). A soft handover is one in which the connection to the target link is established before the serving link is broken (also termed make-before-break). The overlapping period during which the two connections are used in parallel is important to ensure smooth data delivery. Such an approach significantly reduces the possibility of session termination owing to relatively short handover delay and a small number of packet losses when compared to hard handover.

In this paper, we propose a handover scheme called application layer seamless switching (ALSS) to provide smooth multimedia delivery across unicast and multicast networks (Fig. 1). As unicast streaming has a direct relationship between a server and a client whereas multicast streaming is a one-to-many connection, a dedicated multimedia stream should be delivered to the user who is moving from a multicast network to a non-multicast area. During the handover period, ALSS aims to preserve the ongoing multimedia session, i.e., seamless playback. To this end, soft handover forms the basis of ALSS as hard handover is inadequate to support seamless handover of multimedia streams (Cunningham *et al.*, 2009). Here, soft handover refers to one in which the same multimedia stream (but a different delivery method) is sent to the MT via two access points (APs) simultaneously. The multimedia stream from the old AP is terminated only after a specific overlapping period has elapsed. It is, however, not clear how long the delay should be to achieve a seamless playback. This paper is devoted to answering the above question with the consideration of all practical constraints, especially the audio and video buffer behaviors. To demonstrate the realistic effectiveness of ALSS, we built a real-time streaming testbed where a standardized streaming method is adopted. The measurements from such an experimental approach are an important complement to previous analytical and simulation studies. A preliminary version of this paper appeared in Choong *et al.* (2011), which, however, lacked thorough analysis on performance evaluation of user perceived quality and did not consider the possibility of an audio buffer under run, which could hamper the user viewing experience.

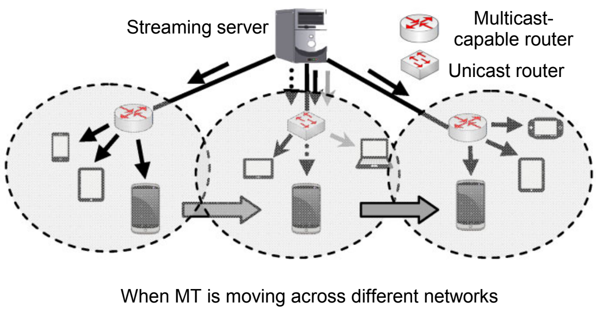


Fig. 1 Handover between multicast and unicast networks

2 Related work

Recently, service providers have shifted their focus from quality of service (QoS) to quality of experience (QoE) (Piamrat *et al.*, 2010). In contrast to QoS, which measures technical parameters such as packet loss and throughput, QoE is an overall satisfaction of a service as perceived subjectively by the end-user. In the context of multimedia session handover, continuous multimedia playback is one of the most important QoE metrics. Armed with such a goal, there has been much research effort on seamless handover for multimedia services. Generally speaking, these handover related works can be divided into two major categories: simulation approach and emulation approach.

A simulation-based handover scheme is the most commonly employed approach found in the literature. The reason is that it is quick and cost-efficient to evaluate the performance of the proposed scheme under different targeted deployment scenarios given the difficulty in testing on a real large-scale commercial network. Quadros *et al.* (2013) proposed a QoE-aware handover scheme to support seamless mobility for multimedia applications. In this scheme, a video quality estimator that maps video characteristics and network impairments to a predicted mean opinion score (MOS) helps make handover decisions. Tong and Yang (2008) proposed a buffer control scheme to facilitate a seamless video handover in a wireless local area network (WLAN) environment. The simulation results show that the deployment of a few-hundred-millisecond video buffer is sufficient to guarantee a seamless handover. The above approaches, however, do not support the handover between unicast and multicast networks. Work that does consider such handover scenarios can be found in Xu *et al.* (2014). However, the detailed steps of describing the video traffic flow during handover for seamless video continuity were not presented. Despite significant gains in terms of handover delay and the packet loss ratio, it is difficult to demonstrate the abilities of discussed prior works in achieving real-time seamless multimedia playback in real systems.

For this reason, several recent contributions have focused on building a testbed to closely emulate the realistic behavior of the actual network. Such approaches belong to the group of emulation approaches which give better insight into the handover performances via experimental evaluation. Cunningham *et al.* (2009) proposed a mechanism that considers levels of network congestion to enable seamless handover of streamed IPTV in a WLAN. However, simply emulating video traffic with a generic data stream generated by Real-time Transport Protocol (RTP) packets is insufficient to capture the real handover effect on streamed video. Due to the unique spatial-temporal characteristics possessed by a video stream, the loss of different video packets induces different levels of distortion. Similarly, Saxena and Roy (2011) emulated the video traffic with a User Datagram Protocol (UDP) based generic data stream in their proposed handover scheme. Politis *et al.* (2010) proposed a QoE-driven handover scheme for both non-scalable and scalable video streaming. Through a QoE rate adaptation scheme, the experimental results show that mobility of video streaming can be better maintained with a scalable video. Lu (2010) proposed a solution using Media Independent Handover (MIH)/ IEEE 802.11 to enable seamless mobility for video streaming sessions. However, the experimental results provide only the handover delay while neglecting any video quality metric. In Xu *et al.* (2013), when one of the interfaces of the MT performs handover, the packets sent to/from the handover interface will be forwarded to their alternative interface in order to ensure continuous reception of packets during the handover process. Again, no evaluation of video quality was given. While this paper follows the emulation approach in which real multimedia streaming is considered, we provide frame-by-frame video quality and an objective-based QoE metric to confirm the effectiveness in terms of playback smoothness.

Another major drawback of all the aforementioned schemes is that they focus on video buffer behavior and simply neglect the audio part of a multimedia bitstream. More specifically, the audio stream is not transmitted over networks in the simulation or emulation studies. In fact, these two streams should be played back in synchronization with each other. If either one of audio and video buffers does not have enough buffered data, a buffer underflow arises at the receiver. This issue becomes especially relevant in the presence of a handover event where packet losses are frequently observed. Even in the case of zero packet losses, if the first few arriving packets from the new network contain only one media (audio or video) datum, poor user viewing experience could still happen. The work presented here takes into account the presence of both audio and video packets from the candidate network during the handover process.

3 Application layer seamless switching

3.1 System architecture and components

Fig. 2 shows the architectural overview of both the client (MT) and server of ALSS. Basically, it consists of two layers. The top layer consists of the proposed modules (shaded blocks) that we have built to manage seamless switching. The underlying modules are standard modules that perform audio/video (A/V) encoding, decoding, and streaming of multimedia data.

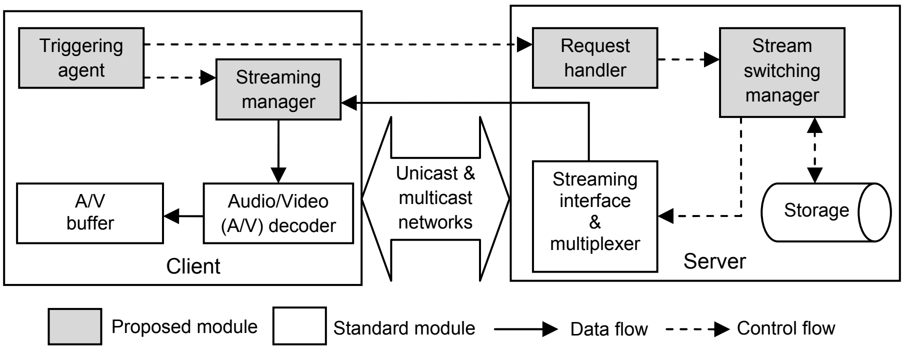


Fig. 2 The client and server architecture for application layer seamless switching (ALSS)

With regard to client operation, there are four components: triggering agent, streaming manager, A/V decoder, and A/V buffer. The triggering agent is responsible for initiating a connection switching and hence in charge of creating an alternative connection to the server during handover. Thus, the proposed handover scheme belongs to the mobile-controlled handover. Depending on the type of alternative connection (unicast/multicast), the triggering agent will either pass over the current streaming's uniform resource locator (URL) or issue a multicast join request. Note that deciding when to trigger a handover goes beyond the scope of this paper. Instead, we focus on the experimental aspect of the handover execution phase for seamless playback experience. For this reason, the switching process is manually triggered with a button pressed on a multimedia player user interface. The streaming manager is to receive streaming contents (both current and new streams) from the server. For the new stream, it extracts payload from the incoming packets in order to distinguish between audio and video packets and to forward the packets to an A/V decoder. The A/V decoder is responsible for decoding audio or video packets, i.e., to assemble defragmented packets into A/V frames and later store them into its corresponding A/V buffer for playback purpose.

Corresponding to the client are four modules at the server: request handler, stream switching manager (SSM), storage, and the streaming interface and multiplexer. The request handler is an active module that is always listening to any incoming client request. Its main task is to extract the IP address of any incoming client and forward it together with the streaming URL to the SSM. SSM uses the streaming URL to look up the resource that is currently being delivered to the requesting client, and then retrieves the current playback time of the sending stream. After that, it accesses the same multimedia from the storage, jumps to the specific playback time frame, and informs the streaming interface to stream the multimedia back to the client using an alternative delivery method. The multiplexer is responsible for multiplexing audio and video streams to form a single stream before transmitting them over the network.

3.2 Switching interaction

Fig. 3 describes the flow of the switching connection. Two phases are considered, namely multicast-to-unicast (M2U) and unicast-to-multicast (U2M), with a scenario as follows: First, the MT connects to a network via its first network interface (NI1) to watch a streaming multimedia with multicast delivery. Playback commences after a sufficient amount of multimedia data (as defined by a target buffer level) has been buffered at the A/V buffer. Whenever a switching is triggered, the MT will send a message to the server through its second network interface (NI2) for requesting a unicast stream. The server then retrieves the URL of the multicast session currently being played and creates a unicast stream for the same multimedia back to the client. To synchronize both multicast and unicast sessions, the server accesses certain parts of the multimedia file and streams a unicast session starting from the current playback time of the multicast session.

When the unicast data packets arrive at NI2, the client will examine whether it is an audio or video stream. The duration from the switching trigger till

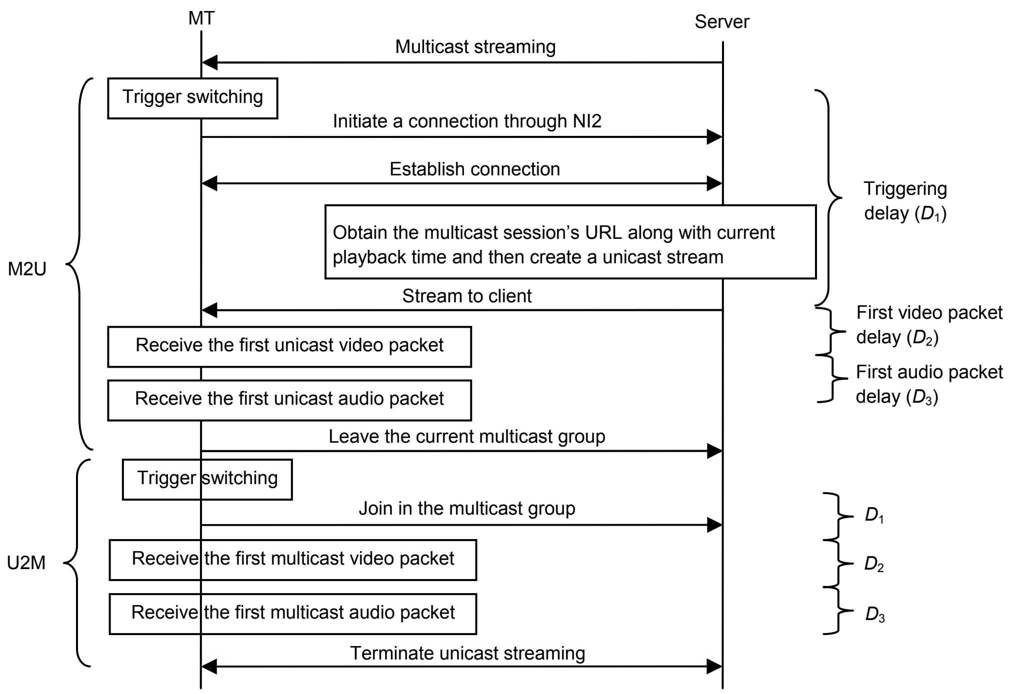


Fig. 3 Interaction between the application layer seamless switching (ALSS) client and server

the receiving of the first packet from NI2 is known as the triggering delay (D_1). The examination process shall continue until the first video unicast packet is captured. This is because in the process of packetizing and multiplexing the audio and video frames into a single data stream by the server, it is possible to pack several audio packets much earlier than any video packet since the video packet requires a longer time to process. In the absence of a video packet, the client video decoder will malfunction due to video buffer starvation, resulting in the occurrence of blank screens during playback. In the worst case scenario, re-buffering will take place until the A/V buffer reaches its target buffer level, again before any playback. The duration from the first received audio packet until the first received video packet is known as the first video packet delay (D_2).

Upon detection of the first unicast video packet, the first and subsequent data packets are forwarded to the A/V decoder. At the same time, the client detects the first audio packet. It is important for the client to stay in the current multicast group until the next (first) unicast audio packet is received. Otherwise, the audio decoder will also malfunction due to audio buffer underflow. Under such circumstances, the video/image may freeze and will not recuperate. For this reason, the current multicast stream is still required to supply audio packets until the next unicast audio packet arrives. Such a duration is defined as the first audio packet delay (D_3). For a seamless handover experience, the handover overlapping period must be at least equal to the sum of D_2 and D_3 where both NI1 and NI2 are receiving data packets.

Continuing our scenario, the next handover is from unicast back to multicast delivery. The client shall first connect to the alternative network by subscribing to the specific multicast group before terminating the existing unicast stream. Similar to the M2U handover, the client shall wait for the first video and first audio packet before terminating the unicast stream.

The three different delays described above are labeled as shown in Fig. 3 (on the right). These delays apply to both M2U and U2M scenarios. The overall handover delay for the connection handover can be defined as the sum of D_1, D_2, and D_3.

4 Testbed setup

We have built a real-time streaming testbed (Fig. 4) to experiment with our handover approach.

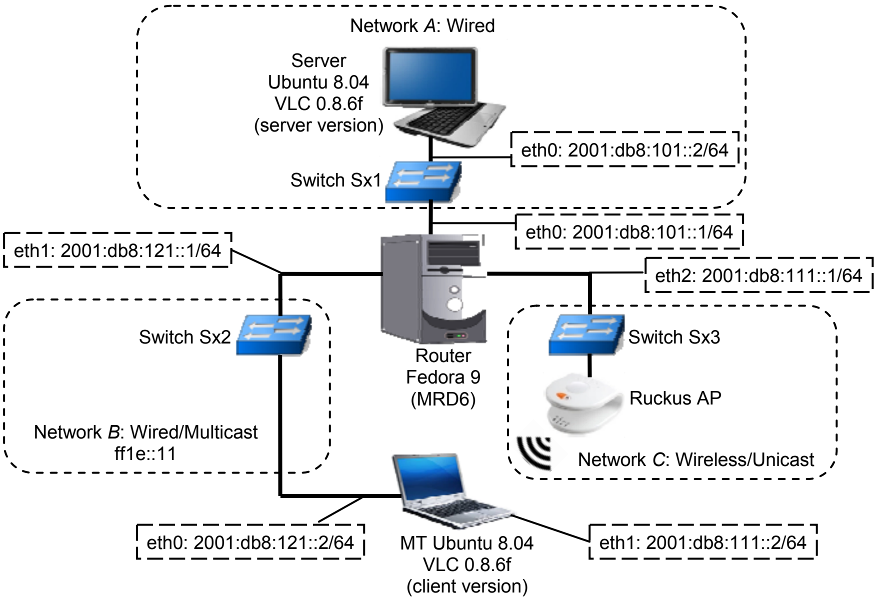


Fig. 4 Testbed for seamless switching between unicast- and multicast-delivered multimedia bitstreams

The testbed is generally divided into three network segments, with network *A* acting as the server segment, and networks *B* and *C* serving as the client segments. Each network is assigned a different static IPv6 address.

Two laptops (one for the server and the other for the client with two network interfaces) and one desktop computer are used, each assigned a static IPv6 address. The operating systems (OS) of the two laptops are both Ubuntu 8.04. The desktop computer acting as the router is installed with MRD6 in Fedora 9 OS with three network interfaces linking all three network segments. MRD6 is an IPv6 routing daemon with multicast forwarding capabilities. The laptop serving as the client is equipped with both Ethernet (802.3) and WLAN (802.11) interfaces, connecting to both networks *B* and *C* respectively. The Ethernet is chosen to mimic other cellular networks as we do not have base station equipment. Network *C* serves as the Wi-Fi access with multicast-capable AP.

The three proposed ALSS modules (except the streaming manager) are integrated into the VLC media player of version vlc-0.8.6f to play the role of a streaming server and client. The remaining module, streaming manager, is implemented with tcpdump, which is a network monitoring tool that captures and analyzes the contents of packets on a specific network interface. This empowers us to identify audio and video packets from the new multimedia stream.

We adopt the MPEG-2 Transport Stream (TS) to deliver multimedia contents in our experiment. MPEG-2 TS is a popular format for transmission of multimedia streams over networks such as IPTV (ISO, 1996). In MPEG, a multimedia stream typically consists of two elementary streams (audio and video). An MPEG encoder converts each elementary stream into its corresponding packetized elementary stream (PES) packets, each carrying either an audio or a video frame. Each PES packet is further split into multiple fixed-length TS packets for transmission over the IP based network. To this aim, audio and video TS packets are multiplexed and encapsulated in the UDP packets, each carrying seven 188-byte TS packets (Fig. 5). The number of audio and video TS packets for a particular UDP packet is allocated in such a way that both the audio and video streams are played back in synchronization with each other.

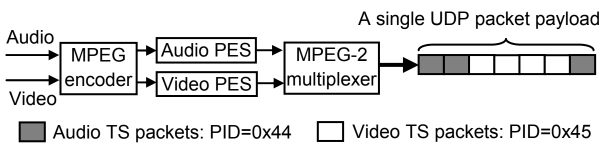


Fig. 5 An example of MPEG-2 Transport Stream

At the receiver side, the UDP payload is demultiplexed into individual PES streams by the packet identifier (PID) values before forwarding them to the appropriate A/V decoder and buffer. If either of the audio and video buffers does not have enough buffered data, a buffer starvation arises at the receiver. Because a single video frame typically consumes more data packets to be displayed as compared to a single audio frame (at least in the multimedia files we experiment with), we privilege the arrival of the video packet to avoid re-buffering. Thus, it is of utmost importance to recognize the values of D_2 and D_3.

5 Experiments and performance evaluation

5.1 Overview

To verify the robustness of ALSS, we tested three multimedia (A/V) files of video with different variable bit-rates and audio with a constant bit-rate (Table 1). For all the three cases, the target buffer level was set to 300 ms.

Fig. 6 visualizes the number of incoming packets from both the wired and wireless interfaces during the overlapping period in KSysGuard. By monitoring the number of packets, it was verified that the multimedia stream has successfully switched from one network to the other, without any interruption on the visual playback. We provide both quality of service (QoS) and objective-measured quality of experience (QoE) indicators in the following. The former measures the handover delay and peak signal-to-noise ratio (PSNR) while the latter estimates the subjective video quality based on video playback errors and PSNR.

5.2 Handover delay

Two approaches were adopted to examine the handover delay. The first approach is by putting several timestamps at different parts of the VLC source

Fig. 6 Overlapping of incoming packets during handover

code to capture the three delay values during the connection handover as explained earlier. In the second approach, a lower (network) layer approach using Wireshark was adopted to further capture and verify packets that flow in and out through both the wired (eth0) and wireless (eth1) network interfaces. Fig. 7 shows different test points using these two approaches, being an extended version of Fig. 3 with the addition of the network interface and router. In short, the various intervals between the neighboring test points define the three different delays as explained earlier. The definition for each time interval is as follows:

M2U: $A \rightarrow B$: D_1; $B \rightarrow C$: D_2; $C \rightarrow D$: D_3.
U2M: $F \rightarrow G$: D_1; $G \rightarrow H$: D_2; $H \rightarrow I$: D_3.

In a similar fashion, we obtained various test points measured with Wireshark, with the addition of test points d, e, i, and j. The intervals of $d \rightarrow e$ and $i \rightarrow j$ refer to the current stream's leave latency. In the context of M2U, the delay $d \rightarrow e$ is called group leave latency, defined as the time from the last listening nodes on a subnet leaving the group to the time no more multicast traffic is forwarded to that subnet. On the other hand, delay $i \rightarrow j$ is the current stream's leave latency for U2M, which is the duration during which the client requests the server to stop sending a unicast stream till the last unicast packet is received.

Table 1 Properties of the three test multimedia files

Multimedia	Video frame rate (frame/s)	Number of video frames	Average video bit-rate (kb/s)	Audio frame rate (frame/s)	Audio bit-rate (kb/s)
I	15	2232	411	38	128
II	25	5353	770	38	112
III	30	2604	4025	38	160

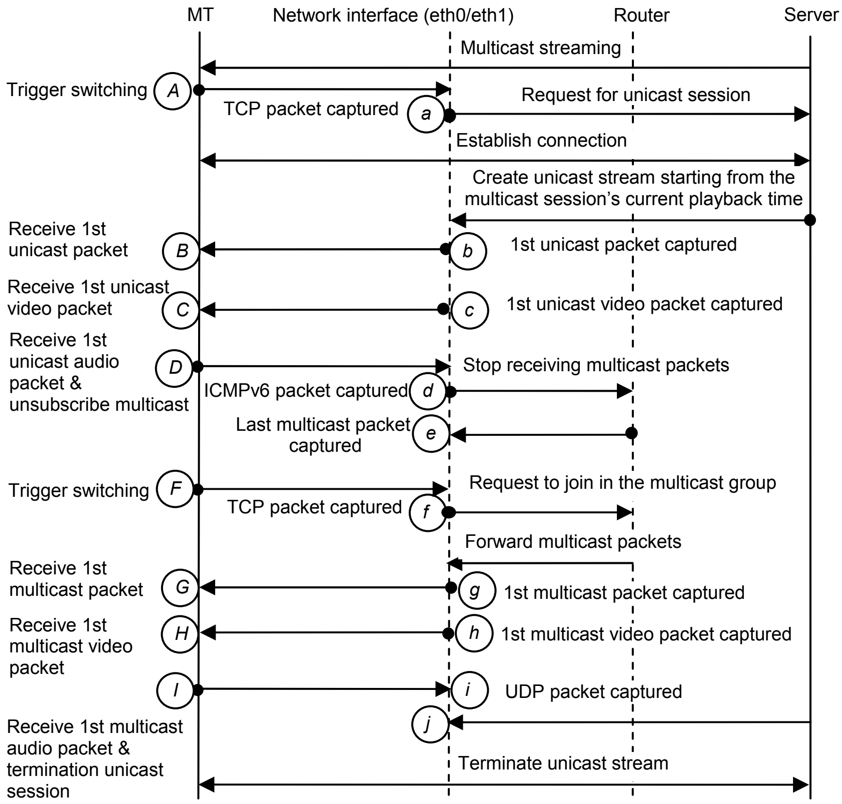


Fig. 7 Sequence diagram for handover

In addition to studying the various delay values during connection handover execution, we compare the overall handover delay of M2U for the system with and without the use of ALSS. The execution flow for M2U without ALSS is as follows:

1. The server streams a multicast session to the client over the Wi-Fi AP.

2. After a while, the Wi-Fi AP is turned off to simulate connection breakdown.

3. Once the client detects that there is no more incoming packet, it then makes a unicast connection to the server over an alternative network interface.

4. The duration for waiting the first unicast packet (both audio and video) is then recorded.

5.3 Handover effect on streamed video

5.3.1 Peak signal-to-noise ratio (PSNR)

PSNR analysis was chosen to study the handover impact on the streamed video as it has often been used in the literature to study video quality. In short, PSNR is calculated by comparing every pixel in the first frame of the streamed video with the corresponding pixel in the first frame of the pre-encoded video, similar for the subsequent frames. PSNR is defined as follows:

$$\mathrm{PSNR} = 10\lg\frac{(\mathrm{MAX_I})^2}{\mathrm{MSE}}, \tag{1}$$

where $\mathrm{MAX_I}$ is the maximum luminance with a value of 255 for a picture coded with 8-bit resolution, and MSE is the mean squared error. MSE is null if two frames being compared are the same. Note that if there is no distortion, the PSNR value should be infinity, according to Eq. (1). For simplicity, we adopted the same approach as in Chan *et al.* (2010) to define the highest value of PSNR as 100 dB. The higher the PSNR value, the higher the received frame quality and the higher the level of viewing satisfaction experienced by the users.

Before studying the handover effect, we examined whether video streaming without handover operation causes any distortion. For this purpose, experiments without handover were conducted for all three videos. The video sent from the server and the video received by the client were extracted into frames using ffmpeg. Then, PSNR was computed in sequence using ImageMagick. Such PSNR results also served as the baseline for comparison with video quality after handover. This will be further described in the following.

For the M2U experiment, the PSNR calculation was slightly different from the method just described. The reason is that the above method assumes no skipping or redundant frames in the streamed video. However, video playback errors such as frame losses and frame redundancy are prevalent in the case of handover due to imperfect synchronization between current and new multimedia streams. A missing or repeated frame would cause inaccurate frames to be compared in PSNR analysis. In this case, temporal alignment is useful as a pre-processing step for frame matching (Chan *et al.*, 2010). It is the process of adding or removing frames from streamed video so that both pre-encoded and streamed videos have the correct number of video frames to be compared.

To this aim, we computed PSNR as follows: First, use ffmpeg to extract the video coding statistics from both pre-encoded and streamed videos and then store them into a separate file. Fig. 8 shows the detailed video coding statistics of each frame, including eight properties: frame number (frame), video quantizer scale (q), frame size (f_size), accumulated frame size (s_size), presentation time stamp (time), bitrates (br), average bitrates (avr_br), and picture type from a group of pictures (GOP) structure (type). Second, we can spot that the highlighted frames in Fig. 8a were actually the lost frames as these frames were missing from Fig. 8b by checking on the frame size. Similarly, repeated frames can be detected by observing those frames with matching frame size. If dropped frames were detected, copies of the previous frame were inserted into that frame position. On the other hand, any repeated frame was simply deleted. Finally, we can compute the PSNR of the streamed video as in the case of video streaming without handover operation.

5.3.2 Video playback smoothness

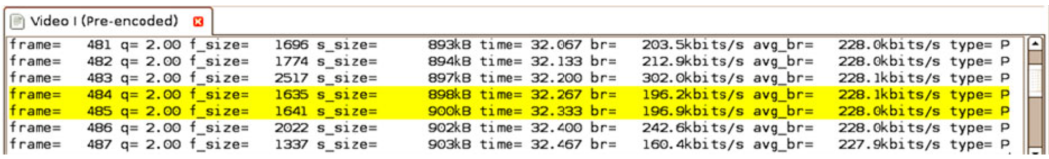
To further establish the effectiveness of the proposed scheme, we adopted another metric to measure video quality, as PSNR does not correlate well with perceived video quality although it is the most widely used objective video quality metric (Wang *et al.*, 2004). Subjective assessment is a more reliable means to determine the end-user's perceived smoothness of video playback. The mean opinion score (MOS), an international standard of multimedia subjective testing, can be used to measure and quantify end-user's perception of video quality (ITU, 2002). MOS is generated by averaging the quality ratings of all individual participants. The rating is shown in Table 2.

However, the MOS method has two major limitations; i.e., it is time consuming and expensive as it involves hiring experts for the assessment. In this

Table 2 Mean opinion score (MOS) (ITU, 2002)

MOS	Impairment	Comment
5	Imperceptible	Excellent
4	Perceptible but not annoying	Good
3	Slightly annoying	Fair
2	Annoying	Poor
1	Very annoying	Bad

(a)



(b)

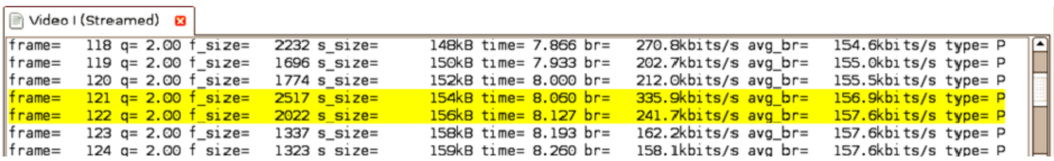


Fig. 8 Identifying extra and lost frames: (a) pre-encoded video I; (b) streamed video I

regard, we have adopted another quick and cost-efficient method called the video gross error detector (GED) to estimate an average end-user's perceived video playback smoothness (Younkin and Corriveau, 2008). The video GED maps the objective metric in terms of video playback errors to the MOS score by using the following equation:

$$\mathrm{MOS} = -0.571\ln\left[\left(\frac{\text{total number of frames}}{\text{frame count}}\right) \times \text{total number of frame errors}\right] + 4.6836, \tag{2}$$

where the total number of frame errors is the sum of the numbers of dropped and repeated frames. Such a mapping is reasonable as QoE measurement should include objective metrics to derive global QoE ratings (Brooks and Hestnes, 2010).

6 Experimental results and discussions

6.1 Handover delay

We have conducted 10 rounds of tests to tackle the issues of deviation, and the average handover delay is reported in Figs. 9–13.

6.1.1 Handover delay at the application layer

Figs. 9 and 10 show the time taken to perform M2U and U2M, respectively, with internal timestamps in VLC source code. Fig. 9 shows that delay *A→B* is almost the same for multimedia I and II, whereas for multimedia III, it is 1 s higher. Two factors may lead to this disparity, i.e., network conditions and the processing capacity of the server. To pinpoint the particular reason, the round-trip time (RTT) of the Transmission Control Protocol (TCP) request was determined. The test results indicated that the RTT is relatively consistent in all three scenarios, which is 0.01 s. This implies that the network conditions for all three cases were stable and this factor could be ruled out for the reason why the first packet of multimedia III arrived lately. The possible reason is the time spent on creating a unicast stream, which includes reading the multimedia file from the storage, jumping to the specific playback time frame, and streaming the data to the client. In general, the larger the content size, the longer it will take for the server to establish a unicast stream.

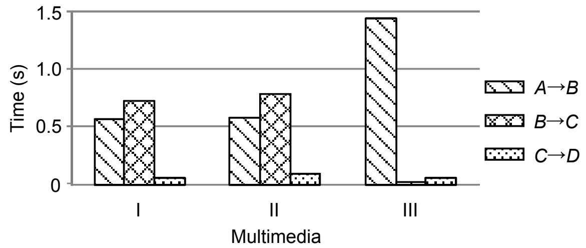


Fig. 9 The time needed to perform multicast-to-unicast (M2U) at the application layer

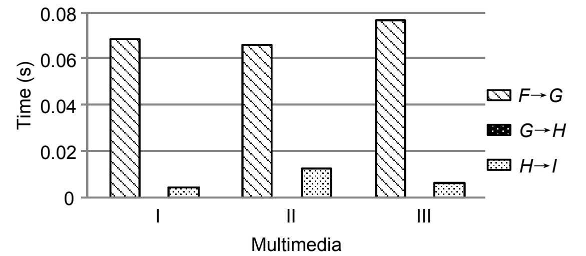


Fig. 10 The time needed to perform unicast-to-multicast (U2M) at the application layer

Fig. 9 also shows a significant difference in delay *B→C* of multimedia I and II as compared to multimedia III. For multimedia I and II, the unicast audio packets reach earlier than the unicast video packets by approximately 0.75 s, while it is 0.003 s for multimedia III. This may be due to the variation between audio and video frame rates. For example, based on Table 1, one second of multimedia I contents requires 38 audio frames and 15 video frames, which implies that to display one video frame, the A/V buffer may need [(1/15)/(1/38)]=2.5 audio frames to be ready. In contrast, for multimedia III, it needs only [(1/30)/(1/38)]=1.3 audio frames to be ready together with one video frame. Hence, the encoder and multiplexer could exhibit synchronization issues at the beginning when it receives a request to create a unicast session for multimedia I and II. The consequence of this issue is that there may be a silence gap for audio during playback. This is due to the design as explained in Section 2.2 that starts accepting a new multimedia stream only after the arrival of the first video packet in order to prevent re-buffering. Nevertheless, such audible glitches should have less impact on user experience when compared to a video re-buffering. Besides that, delay *C→D* is relatively consistent for all three multimedia, which is approximately 60 ms. As expected, multimedia III with no synchronization issue has the least overlapping period (D_2+D_3), which is 56 ms.

As shown in Fig. 10, the three multimedia have approximately the same delay $F \rightarrow G$ with an average value of 0.07 s. Meanwhile, delay $G \rightarrow H$ is zero and delay $H \rightarrow I$ is consistently small for all three multimedia since the existing multicast session is a well-synchronized multimedia stream, which means the first UDP packet received by the client carries both the audio and video packets. In addition, multimedia I has the least overlapping period of only 4 ms.

6.1.2 Handover delay at the network layer

Figs. 11 and 12 show the time taken to perform M2U and U2M, respectively, with packet analysis done using Wireshark. As shown in Fig. 11, delays $a \rightarrow b$, $b \rightarrow c$, and $c \rightarrow d$ are consistently similar to delays $A \rightarrow B$, $B \rightarrow C$, and $C \rightarrow D$, respectively, due to the same measured parameters. However, although delay $c \rightarrow d$ is almost the same for all three multimedia, there is a significant difference in the number of received packets during that interval (Table 3). For multimedia I and II, the number of received packets is about five times more than the number of received packets for multimedia III, which is unusual since multimedia III contains the highest bit-rate data. This is due to the unsynchronized multimedia stream. As discussed earlier, the audio packets are sent earlier than the video packets for multimedia I and II. This could lead to unsynchronized playback where only the audio frame is available while the video packet is absent. In response to this, we believe that the streaming interface and multiplexer transmits the video packets of multimedia I and II at a higher data rate to ensure that there is no skipping of video frames due to the late arrival of the video packets. After that, it will resume to a normal streaming rate to form a synchronized stream.

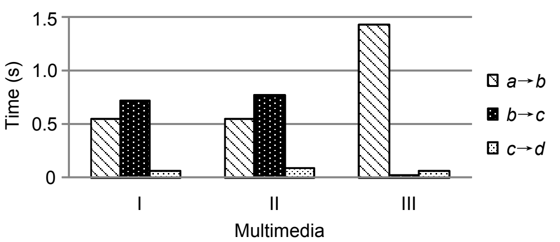


Fig. 11 The time needed to perform multicast-to-unicast (M2U) at network and transport layers

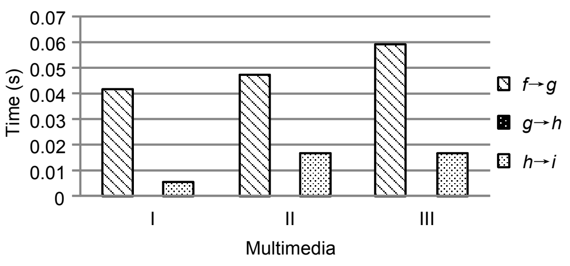


Fig. 12 The time needed to perform unicast-to-multicast (U2M) at network and transport layers

As shown in Fig. 12, delays $f \rightarrow g$, $g \rightarrow h$, and $h \rightarrow i$ are almost similar to delays $F \rightarrow G$, $H \rightarrow I$, and $I \rightarrow J$, respectively, due to the same measured parameters. Delay $f \rightarrow g$ is called the multicast join latency, which is too small compared to $a \rightarrow b$.

Table 3 Total received packets during delay $c \rightarrow d$

Multimedia	Number of received UDP packets
I	135
II	155
III	25

6.1.3 Overall discussion

Recall that there is only one MT in our testbed. Therefore, the MT is considered as the only (last) member on its associated subnet. When the router receives a multicast leave request, it needs 4 s of delay $d \rightarrow e$ to check if there is still any multicast subscriber in the network before it stops forwarding the multicast traffic to the subnet. On the other hand, for U2M, the server needs 3.75 s of delay $i \rightarrow j$ to confirm whether its recipients are still accepting the unicast data packets for playback. Note that packets received by the MT at intervals $d \rightarrow e$ and $i \rightarrow j$ will not be buffered for playback. Hence, such group/session leaving delay will not contribute to the entire handover delay.

Fig. 13 shows the overall handover delay of U2M and M2U. The U2M handover is faster than the M2U handover by 1.34 s. This is due to the readily available and well-synchronized multicast stream. More specifically, the MT joins in an already existing multicast group by requesting the router to

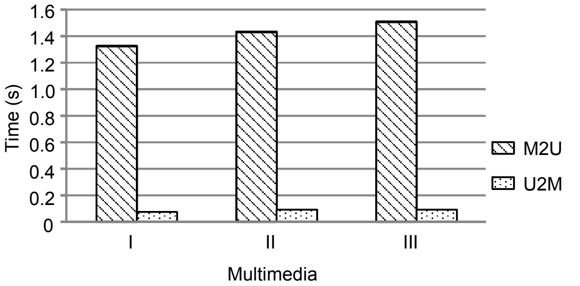


Fig. 13 Overall handover delay for multicast-to-unicast (M2U) and unicast-to-multicast (U2M)

forward data packets to it. On the contrary, a unicast stream is created upon the request from the MT.

Fig. 14 compares the overall handover delay of M2U for a system without and with the use of ALSS for the delivery of multimedia I. A disconnection period of 5 s for M2U without ALSS is observed (Fig. 14a), causing underflow at the A/V buffer, which leads to an interrupted playback. With a target buffer level set to 300 ms, the total waiting time is around 5.3 s before playback is resumed. On the contrary, with the use of ALSS, smooth playback is observed (Fig. 14b) because the decoder always has suffcient audio and video frames for playback. Note that both D_2 and D_3 must take place during the overlapping period to ensure seamless handover experience.

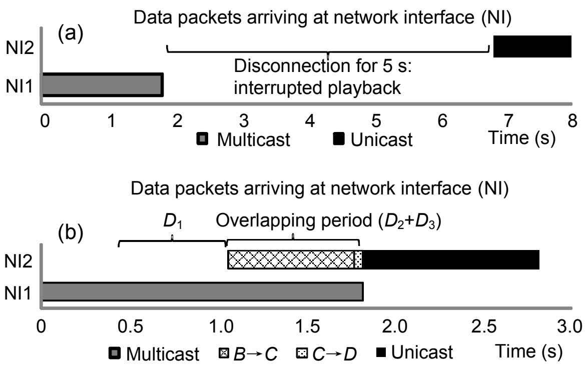


Fig. 14 Performance comparison of M2U without (a) and with (b) ALSS for multimedia I

6.2 Handover effect on streamed video quality

6.2.1 Peak signal-to-noise ratio (PSNR)

Fig. 15 compares the frame size of pre-encoded and streamed videos for videos I and III. The frame number plotted here refers to the frame number of the streamed video (refer to the first column of the table in Fig. 8b). From Fig. 15a, visual inspection of the solid circle suggests that two frames of the streamed video I appear earlier than the pre-encoded video I due to lost frames. By tracing backward, we found that the issue started off at frame 122, as indicated by the dotted ellipse. In a similar fashion, the frames of streamed video III appear later than the pre-encoded video III due to duplicated frames (Fig. 15b). The number of lost or duplicated frames is tabulated in Table 4.

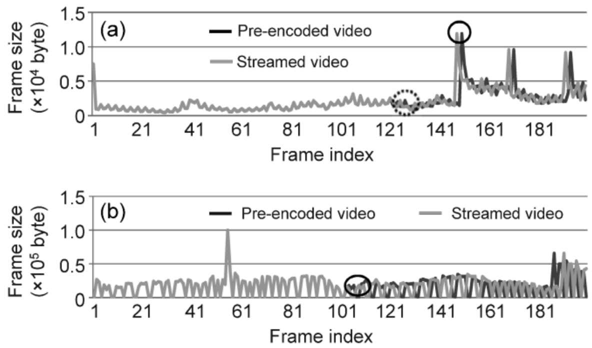


Fig. 15 Comparison of frame size between pre-encoded and streamed videos: (a) video I; (b) video III

Table 4 Number of lost/duplicated frames

Video	Number of lost frames	Number of duplicated frames
I	2	0
II	0	1
III	0	4

Fig. 16 displays the PSNR values of all three videos with and without M2U handover for the selected 100 frames, showing significant PSNR differences. As expected, for all three videos without handover, the resulting average PSNR is 100 dB. This means that there are no distortions in any frame of these videos. If there is any frame pair returning a PSNR not equal to 100 dB, we can conclude that there is distortion caused by the handover.

For each streamed video, the first frame pair returning a PSNR not equal to 100 dB is selected as the starting point of the graph. Fig. 16 shows that video II obtains the highest average PSNR among all videos at 97.76 dB with five distorted frames. The average PSNR for video I is 84.02 dB with 26 distorted frames, while the average PSNR for video III declines by 36.3% as compared to video I with the highest number of distorted frames. Obviously, this indicates that video III has the lowest QoS during connection handover. The reason lies in the nature of the predictive video coding technique, more specifically, the group of pictures (GOP) structure which results in error propagation (Kiraly *et al.*, 2010). A GOP is a group of successive frames reflecting spatial motion activities in video shots. It always begins with an I-frame which does not require any additional information for reconstruction and then a P-frame which requires the prior decoding of the preceding I- or P-frame to be decoded. Therefore, if an error occurs within a GOP (e.g., due to frame loss), the error will propagate till the next reference picture.

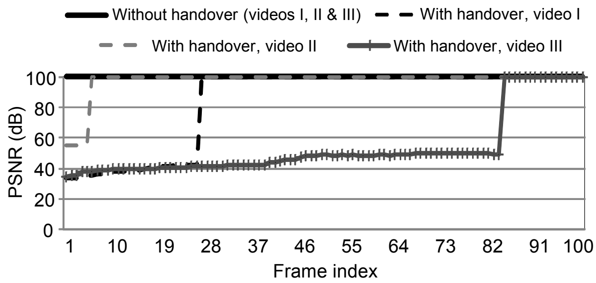


Fig. 16 PSNR for streamed video with and without handover

To verify the stated reason, ffmpeg was again used to find the GOP length for each streamed video, as tabulated in Table 5. Here, the GOP length refers to the distance (in number of frames) between the first frame after lost or duplicated frames and the next I-frame. For example, as shown in Figs. 8b and 17, the first frame after lost frames and the next I-frame for video I are frame 122 and frame 148, respectively. Hence, the GOP length is 148−122=26, which is the same as the number of frames returning PSNR not equal to 100 dB, as shown in Fig. 16. This means that the MT must wait for 26 future frames to arrive before a possible correction with the next I-frame is received.

6.2.2 Video playback smoothness

Using Eq. (2) and information found from Tables 1 and 4, we estimated the MOS (Table 6). All three streamed videos, especially the first two, have very good estimated video playback experience in terms of smoothness. It is, however, not clear how the frame-by-frame quality is perceived by an end-user. To this aim, we map the PSNR of streamed video to MOS by following the recommendation in ITU-T (Table 7). As the minimum PSNR found in Fig. 16 is 33.5 dB, it can be further ensured that the overall streamed video quality is at least at a 'Good' level during the handover process.

Table 5 GOP length of the three test videos

Video	GOP length
I	26
II	5
III	84

Table 6 Estimated mean opinion scores (MOS)

Video	Estimated MOS
I	4.2878
II	4.6836
III	3.8920

Table 7 PSNR to MOS mapping (ITU-T, 2008)

MOS	PSNR (dB)	Comment
1	<20	Bad
2	[20, 25)	Poor
3	[25, 31)	Fair
4	[31, 37)	Good
5	≥37	Excellent

7 Conclusions

We have proposed a handover scheme called ALSS by taking into account both audio and video buffer behaviors. The goal is to provide a seamless handover between unicast and multicast networks

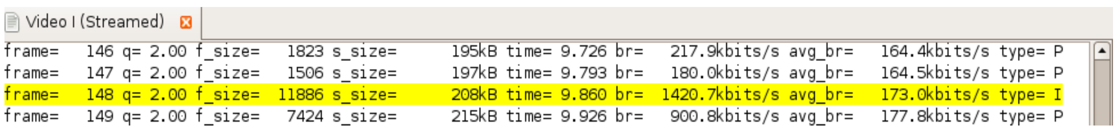


Fig. 17 Next I-frame for streamed video I

so that the ongoing multimedia session can be preserved. This is accomplished by a soft handover in which the client receives two multimedia streams simultaneously. To establish the effectiveness of ALSS under realistic conditions, we built a real-time streaming testbed and selected the popular MPEG-2 TS as the streaming method. We then adopted several open source tools to measure the performance evaluation metrics and converted these objective measures into estimated subjective metrics. Experimental results showed that the overlapping period took a minimum of 56 and 4 ms for M2U and U2M handover, respectively. The frame-by-frame quality of the streamed video was categorized at least as 'Good' based on ITU-T recommendations. The estimated MOS scores confirmed the video playback smoothness.

References

Åström, B., Edlund, P., 2009. Switching Between Delivery Methods in an IPTV Communication Network. US Patent 20110067081 A1.

Brooks, P., Hestnes, B., 2010. User measures of quality of experience: why being objective and quantitative is important. *IEEE Network*, **24**(2):8-13. [doi:10.1109/MNET.2010.5430138]

Chan, A., Zeng, K., Mohapatra, P., *et al.*, 2010. Metrics for evaluating video streaming quality in lossy IEEE 802.11 wireless networks. IEEE INFOCOM, p.1-9. [doi:10.1109/INFCOM.2010.5461979]

Choong, K.N., Lee, C.S., Tham, M.L., *et al.*, 2011. Prototype implementation of a connection handover system for video streaming with different delivery methods. 17th Asia-Pacific Conf. on Communications, p.794-799. [doi:10.1109/APCC.2011.6152916]

Cunningham, G., Perry, P., Murphy, J., *et al.*, 2009. Seamless handover of IPTV streams in a wireless LAN network. *IEEE Trans. Broadcast.*, **55**(4):796-801. [doi:10.1109/TBC.2009.2030466]

Hilt, B., Sarni, M., Lorenz, P., 2009. Enhancement of channel switching scenario and IGMPv3 protocol implementation in multicast IPTV networks. *Int. J. Adv. Networks Serv.*, **2**(2-3):167-181.

ISO, 1996. ISO/IEC 13818-1:1996. Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Systems. International Organization for Standardization, Geneva.

ITU, 2002. ITU-R Recommendation BT.500-11. Methodology for the Subjective Assessment of the Quality of Television Pictures. International Telecommunication Union, Geneva.

ITU-T, 2008. Recommendation P.910. Telephone Transmission Quality, Telephone Installations, Local Line Network. International Telecommunication Union, Geneva.

Kiraly, C., Abeni, L., Lo Cigno, R., 2010. Effects of P2P streaming on video quality. IEEE ICC, p.1-5. [doi:10.1109/ICC.2010.5502003]

Lu, G., 2010. Enable multimedia mobility with IEEE 802.21. IEEE WoWMoM, p.1-6. [doi:10.1109/WOWMOM.2010.5534987]

Namburi, P., Sarac, K., Almeroth, K., 2006. Practical utilities for monitoring multicast service availability. *Comput. Commun.*, **29**(10):1675-1686. [doi:10.1016/j.comcom.2005.07.012]

Piamrat, K., Singh, K.D., Ksentini, A., *et al.*, 2010. QoE-aware scheduling for video-streaming in high speed downlink packet access. IEEE WCNC, p.1-6. [doi:10.1109/WCNC.2010.5506102]

Pitsillides, A., Christophorou, C., 2007. MBMS handover control: a new approach for efficient handover in MBMS enabled 3G cellular networks. *Comput. Networks*, **51**(18):4897-4918. [doi:10.1016/j.comnet.2007.08.005]

Politis, I., Dounis, L., Dagiuklas, T., 2010. H.264/SVC vs. H.264/AVC video quality comparison under QoE driven seamless handoff. *Signal Process. Image Commun.*, **27**(8):814-826.

Quadros, C., Cerqueira, E., Neto, A., *et al.*, 2013. A quality of experience handover system for heterogeneous multimedia wireless networks. IEEE ICNC, p.1064-1068. [doi:10.1109/ICCNC.2013.6504239]

Saxena, N., Roy, A., 2011. Novel framework for proactive handover with seamless multimedia over WLANs. *IET Commun.*, **5**(9):1204-1212. [doi:10.1049/iet-com.2010.0004]

Tong, S.R., Yang, S.H., 2008. Buffer control to support a seamless stream handoff in a WLAN that employs simulcast streaming. *IEEE Trans. Wirel. Commun.*, **7**(1):260-268. [doi:10.1109/TWC.2008.060477]

Wang, Z., Bovik, A.C., Sheikh, H.R., *et al.*, 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, **13**(4):600-612. [doi:10.1109/TIP.2003.819861]

Xu, X., Cui, Y., Liu, J., *et al.*, 2013. Seamless handover scheme for real-time multimedia services in PMIPv6. CHINACOM, p.834-839. [doi:10.1109/ChinaCom.2013.6694711]

Xu, Y.H., Chow, C.O., Tham, M.L., *et al.*, 2014. An enhanced framework for providing multimedia broadcast/multicast service over heterogeneous networks. *J. Zhejiang Univ.-Sci. C (Comput. & Electron.)*, **15**(1):63-80. [doi:10.1631/jzus.C1300205]

Younkin, A., Corriveau, P., 2008. The effects of quantity and location of video playback errors on the average end-user's experience. IEEE BMSB, p.1-5. [doi:10.1109/ISBMSB.2008.4536625]