

## Research Article

# Numeric Analysis for Relationship-Aware Scalable Streaming Scheme

Heung Ki Lee,<sup>1</sup> Jaehee Jung,<sup>1</sup> Kyung Jin Ahn,<sup>1</sup> Hwa-Young Jeong,<sup>2</sup> and Gangman Yi<sup>3</sup>

<sup>1</sup> Samsung Electronic Co., Suwon, Republic of Korea

<sup>2</sup> Humanitas College, Kyung Hee University, Seoul, Republic of Korea

<sup>3</sup> Department of Computer Science & Engineering, Gangneung-Wonju National University, Gangwon-do, Republic of Korea

Correspondence should be addressed to Gangman Yi; [gangman@cs.gwnu.ac.kr](mailto:gangman@cs.gwnu.ac.kr)

Received 13 March 2014; Accepted 29 April 2014; Published 12 June 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 Heung Ki Lee et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Frequent packet loss of media data is a critical problem that degrades the quality of streaming services over mobile networks. Packet loss invalidates frames containing lost packets and other related frames at the same time. Indirect loss caused by losing packets decreases the quality of streaming. A scalable streaming service can decrease the amount of dropped multimedia resulting from a single packet loss. Content providers typically divide one large media stream into several layers through a scalable streaming service and then provide each scalable layer to the user depending on the mobile network. Also, a scalable streaming service makes it possible to decode partial multimedia data depending on the relationship between frames and layers. Therefore, a scalable streaming service provides a way to decrease the wasted multimedia data when one packet is lost. However, the hierarchical structure between frames and layers of scalable streams determines the service quality of the scalable streaming service. Even if whole packets of layers are transmitted successfully, they cannot be decoded as a result of the absence of reference frames and layers. Therefore, the complicated relationship between frames and layers in a scalable stream increases the volume of abandoned layers. For providing a high-quality scalable streaming service, we choose a proper relationship between scalable layers as well as the amount of transmitted multimedia data depending on the network situation. We prove that a simple scalable scheme outperforms a complicated scheme in an error-prone network. We suggest an adaptive set-top box (AdaptiveSTB) to lower the dependency between scalable layers in a scalable stream. Also, we provide a numerical model to obtain the indirect loss of multimedia data and apply it to various multimedia streams. Our AdaptiveSTB enhances the quality of a scalable streaming service by removing indirect loss.

## 1. Introduction

The motivation for this paper is to provide high-quality multimedia service over mobile networks. In a mobile network, two trends make it difficult to improve multimedia service. First, the introduction of smart phones has dramatically increased the volume of video traffic over mobile networks [1], with video consuming most of the available wireless resources [2]. Second, users expect a high-quality streaming service. Thus additional wireless resources are required to satisfy those users [1].

In this paper, we present a solution for enhancing the quality of streaming services over mobile networks. One solution is to improve the capacities of wired and wireless links between the multimedia streaming server and

the mobile client. However, updating the mobile network infrastructure is too expensive. Even though Internet Service Providers (ISPs) have continued to improve the speed of mobile networks, they cannot satisfy user thirst for high-quality multimedia services.

Another solution is to decrease the error rate of mobile networks. Streaming services over mobile networks deliver media data under error-prone network environments [3, 4]. Also, users of mobile networks compete for limited wireless resources for receiving multimedia streams. Such severe competition dramatically increases the error rate of mobile networks. Therefore, the quality of the streaming service might be reduced by the increased error rate of mobile networks. However, we cannot control cross-traffic from other devices. In this paper we explore a third approach:

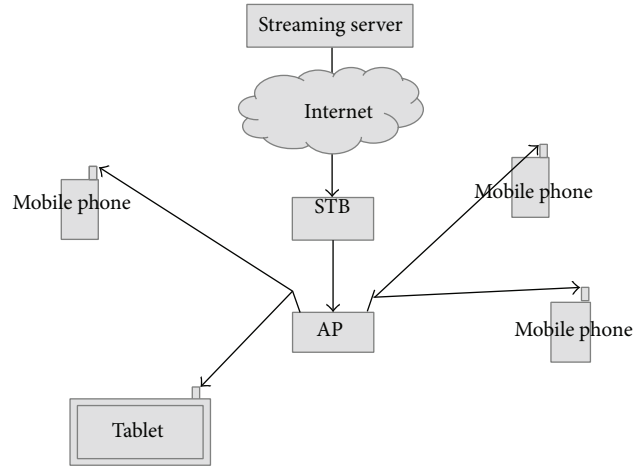


FIGURE 1: Diagram of scalable streaming in a wireless environment.

decreasing the negative effect resulting from losing packets. Error propagation through direct or indirect lost packets worsens the quality of streaming services over mobile networks. As a result, the high quality of media data in mobile networks can lead to frustrating user experience because of frequent data buffering or distorted frames.

To address this problem, content providers (CPs) calibrate streaming and cache servers through a scalable streaming scheme depending on network status. If the CP provides media at different qualities without a scalable streaming scheme, they would need to store all these different media on their own servers and incur costs associated with maintenance of redundant media of different quality. This would increase the cost of maintaining media streams [5–8]. However, a scalable streaming service divides one large media file into several layers. Therefore, by using a scalable streaming scheme, CPs can eliminate redundant stored media data. A scalable streaming scheme consists of a base layer and enhancement layers. The base layer is necessary for decoding; enhancement layers are not themselves decoded but they increase the quality of the streaming service. In a congested network, the mobile node just requests the base layer for seamless streaming. When the mobile network is stabilized, the CP provides all scalable layers including the base layer and enhancement layers to the user. Therefore the user can obtain streaming service with high quality.

Figure 1 shows an example of a scalable streaming service for a mobile network. One set-top box (STB) receives the scalable streams from the streaming server and then forwards scalable layers to mobile nodes including mobile phones and tablets. Usually, the STB is located at one place (e.g., a restaurant, a shop, or a bus station), so the network between the streaming server and the STB is a stable wired network that provides the connection without delay or lost packets during the streaming service. However, the wireless network between the STB and the mobile node is not guaranteed. In wireless networks, several mobile nodes share wireless resources for providing service to mobile users. Interference between mobile nodes can cause the network to drop or delay packets.

Regardless of the benefit of a scalable streaming service, the relationship between layers degrades the quality of the scalable streaming service over an error-prone network. Even though one layer is transmitted successfully, the absence of a reference layer wastes other related layers at the decoder. Therefore, the loss of one packet invalidates its own layer and its referring layer. To improve the performance of scalable streaming services over error-prone networks, we should reduce the dependency between layers, thereby decreasing the amount of related media data for one packet and wasted media data caused by single-packet loss. We suggest an *adaptive set-top box (AdaptiveSTB)* that lessens the dependency between layers transmitted over wireless networks. The *AdaptiveSTB* is located between the wired network and the wireless network and converts complex hierarchical scalable streams into scalable streams consisting of layers with low dependency.

In summary, in this paper we provide the following contributions. We present a service design for an *AdaptiveSTB* that decreases the dependence among scalable layers. Our *AdaptiveSTB* converts the receiving scalable streams with high dependency into scalable streams with low dependency. As a result, it decreases the indirect loss of media data and increases streaming service performance even over mobile networks. We then analyze a media scheme to convert scalable streams. Also we provide a numerical model for showing the amount of multimedia data. Finally, we apply our *AdaptiveSTB* to various streams. In Section 2, we introduce existing adaptive scheme and scalable scheme in detail. Section 3 explains our *AdaptiveSTB* in detail. Section 4 shows experiments for scalable streaming service and results, while Section 5 concludes this paper.

## 2. Related Work

### 2.1. Background

**2.1.1. Scalable Streaming versus Adaptive Streaming.** There are two schemes for adapting the quality of multimedia stream

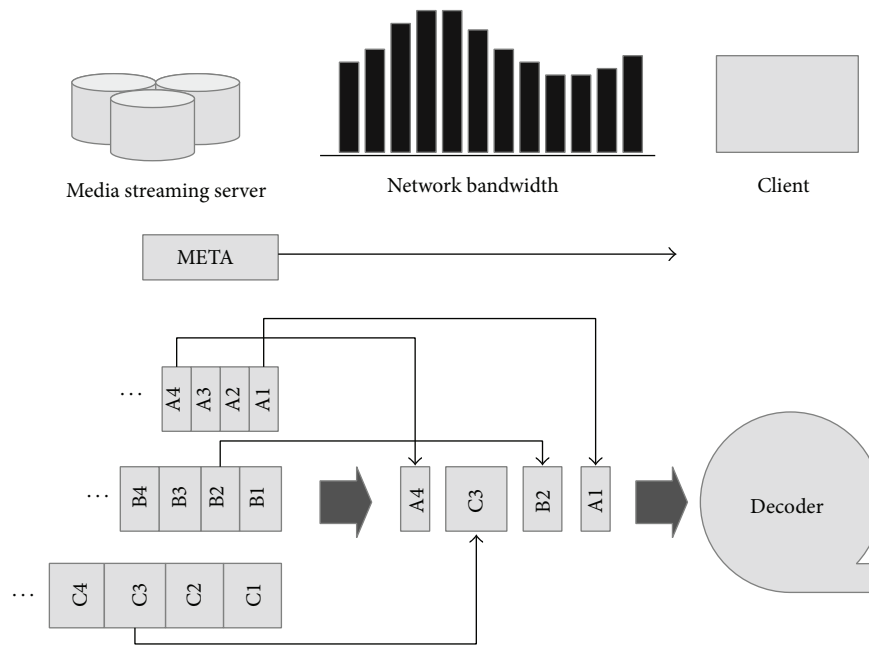


FIGURE 2: Adaptive streaming scheme.

services based on network status: *adaptive streaming* and *scalable streaming*.

In an adaptive streaming scheme, redundant multimedia streams with different quality data are stored in a storage area. Based on the bandwidth, adaptive streaming schemes can switch which stream to send to the user. Figure 2 shows a general adaptive streaming scheme. When the user initiates the adaptive streaming service, the user requests the metafile that describes which streams are available to the user. When the network is congested at the beginning of the streaming service, the adaptive streaming scheme selects the A1 media stream, which has the lowest quality. When the network condition becomes stable, the adaptive streaming scheme switches to the B2 media stream, which is of medium quality. When the network allows for higher quality streams, the user can request the C3 media stream, which has the highest quality. After that, the user changes to media streams with the lowest quality when the network is congested.

Figure 3 shows the scalable streaming scheme. With a scalable streaming service, one stream can be divided into several layers. The base layer can provide the streaming service by itself, but the quality is improved when more layers are included. Upon first use, the user requests the metafile that shows which multimedia streams can be served. The user selects the lowest stream that includes only the A1 base layer. When the network becomes stable, the user selects more layers with medium quality to add B2 enhancement layers to A2 base layers. When the available bandwidth of the network is approved for the highest quality of scalable streaming service, the user requests whole layers including one base layer A3 and two enhancement layers B3 and C3. The user only requests base layer A4 to save wireless resources over a congested network. The scalable streaming service provides the proper quality of the streaming service based on

network conditions. Also, service providers can save space for storing media layers, thereby reducing the cost of maintaining the multimedia system.

**2.1.2. Relationship between Frames.** One media file has various frames, each of which shows one scene in the stream. There are three kinds of frames in the stream: the I frame, the P frame, and the B frame. The I frame contains all the information for showing one scene, whereas a decoder needs to be used to get additional information from other frames for decoding P or B frames. The P frame requires some information from the previous P or I frame, whereas the B frame needs to obtain information from the previous P or I frame and the future P or I frame at the same time.

The hierarchical structure between frames is critical to determining the quality of the scalable streaming service. The relationship between layers determines which layer can be available at the decoder. The scalable stream extracts multiple layers from one stream following each policy. The referring layer cannot be decoded without the reference layer. Therefore, the scalable stream increases the dependency between layers, adding an interframe relationship, thereby complicating the relationship between layers and making them harder to decode.

Figure 4 shows the relationship between layers in *MP4 scalable streaming*. There are several scalable layers: the base layer (Layer 1) and several enhancement layers (Layer 2, Layer 3, and Layer 4). Layer 1 is required for decoding the frame; Layer 2 improves the quality of Layer 1. Therefore, when Layer 1 is not available, Layer 2 cannot be decoded. Also, Layer 3 and Layer 4 are required above the scalable layers for each frame. The I frame can be decoded by itself, but the P frame refers to one previous frame. In Figure 4, only when Layer 1 of

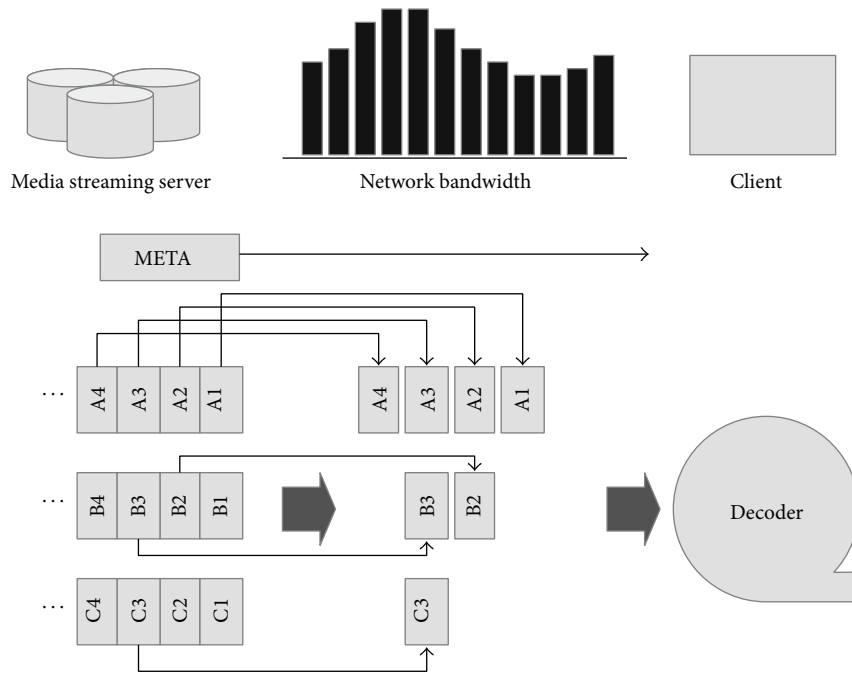


FIGURE 3: Scalable streaming scheme.

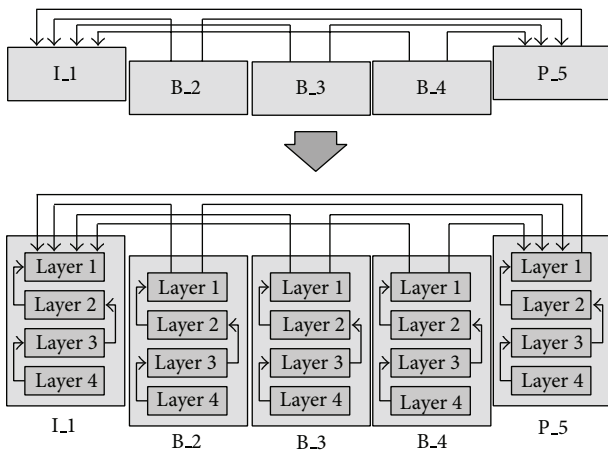


FIGURE 4: MP4 scalable streaming scheme.

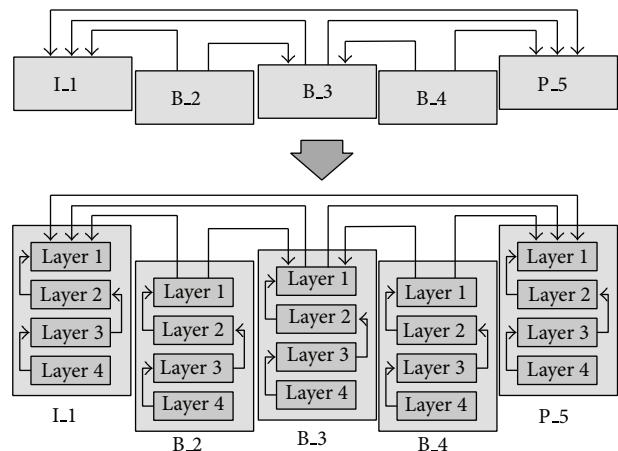


FIGURE 5: H.264 scalable streaming scheme.

the I<sub>1</sub> frame is available can scalable layers of the P<sub>5</sub> frame be decoded. Also, the B frame requires two frames; it needs Layer 1 of the I<sub>1</sub> frame and Layer 1 of the P<sub>5</sub> frame.

In Figure 5, the I and P frames in *H.264 scalable streaming* require no frames or one frame for decoding a frame like MP4 scalable streaming. Also, the relationship between scalable layers at two frames is similar to that in MP4 scalable streaming, but B frames in H.264 scalable streaming require two or more frames to be decoded. In Figure 5, the B<sub>2</sub> frame requires three scalable layers: Layer 1 of the I<sub>1</sub> frame, Layer 1 of the P<sub>5</sub> frame, and Layer 1 of the B<sub>3</sub> frame. This complicated hierarchical structure between frames decreases the network's availability for providing a temporal scalable streaming service.

In this paper, we propose an *AdaptiveSTB* that improves the performance of scalable streaming by reducing the complexity of the relationships between layers. The *AdaptiveSTB* receives the media data from the media server and forwards them to the client through a wireless network. The *AdaptiveSTB* converts the H.264 scalable stream into an MP4 scalable stream before forwarding the cached media.

## 2.2. Previous Work

**2.2.1. Streaming Service.** Numerous schemes have been proposed for handling partial errors in packets. To enhance the quality of a scalable streaming service, [9] increases the availability of the base layer through Multiple Description

Coding (MDC). [1] suggests using the MDC scheme for devices with low computing power and narrow dynamic available bandwidth. Unequal Error Protection (UEP) in layer partitioning has been suggested to improve the performance of streaming in [10]. [11] adjusts the level of Forward Error Correction (FEC) for providing a scalable streaming service. [12] suggests using SoftCast to provide unequal error protection in the video encoding step in wireless networks. In [13], FlexCast selects critical bits of a video through distortion grouping for providing efficient video encoding schemes. [14] suggests a scheme in which a peer device forwards the receiving layers to other devices.

Also, there has been much research on transcoding schemes. ISP proxies, a task dispatcher, and a client provide the transcoding scheme through multiple caching policies in [15]. [16] suggests using Hadoop to conduct a transcoding scheme for a variety of video content suitable under network conditions. In [17], CloudStream is used to enhance the performance through a parallel scheme in transcoding videos. [18] has evaluated the resource demand for a transcoding scheme in various media services.

**2.2.2. Use of Intermediate Nodes for Streaming Video.** For improving streaming service quality, active intermediate nodes have been deployed during streaming [19–22]. When a network is congested, intermediate node degrades quality of the cached stream and then provides it to the mobile node in [19, 23].

In [20, 24], an intermediate node removes the streaming data with large jitter. The intermediate node decides to retransmit the lost packet using the presentation time of the multimedia data in [21]. [22] suggests a scheme in which the intermediate node sends the lost section of multimedia to the user when the user is missing some section in the broadcasting service.

A set-top box is an intermediate node located between the wired network and the wireless network through the streaming service. In [25], the STB consists of four blocks: a Media Codec, a Graphic Module, a Presentation Module, and a Network Module. [26] provided additional functions to the TV STB including video recording and adapting quality of recorded video. [27] detected the lost packets and jitter for improving service to the user. [28, 29] propose using video proxies to increase quality of the streaming service. Also, [30, 31] improve the performance through caching and prefetching strategies.

### 3. Adaptive Set-Top Box

**3.1. Simple Scalable Streaming Service.** The quality of a scalable streaming service is influenced by the dependency among the scalable layers in a scalable streaming service. The hierarchical relationship between scalable layers determines the decoding possibilities for transmitting the packet to the client over the wireless network. When reference frames are not transmitted successfully, the referring frames cannot be decoded. The complicated reference relationships between scalable layers of H.264 streaming increases the possibility

TABLE 1: Scalable streaming conversion variables.

$G_g$	The $g$ th GoV in the stream
$F_{g,f}$	The $f$ th frame at the $g$ th group of video (GoV)
$RF(j, g, f)$	The $j$ th reference frame of the $f$ th frame at the $g$ th GoV
$L_{g,f,k}$	The $k$ th layer of the $f$ th frame at the $g$ th GoV
$P_{g,f,k,s}$	The $s$ th packet in the $k$ th layer of the $f$ th frame at the $g$ th GoV
NG	Number of GoVs in the multimedia stream
$NF_g$	Number of frames at the $g$ th GoV
$NRF_g$	Number of reference frames of the $f$ th frame at the $g$ th GoV
$NL_{g,f}$	Number of layers of the $f$ th frame at the $g$ th GoV
$NP_{g,f,k}$	Number of packets in the $k$ th layer of the $f$ th frame at the $g$ th GoV
PLR	Packet loss rate
$VLI_{g,f,k}$	Validity of the $k$ th layer of the $f$ th frame at the $g$ th GoV
$VLV_{g,f,k}$	Validity of the $k$ th layer of the $f$ th frame at the $g$ th GoV under the relationship between scalable layers in the same frame
$VLH_{g,f,k}$	Validity of the $k$ th layer of the $f$ th frame at the $g$ th GoV in the stream

of discarding the referring frame. Figures 4 and 5 show the relationship between scalable layers in a scalable streaming scheme. In MP4 scalable streaming, whole B frames require two frames, including an I frame or a P frame. Even though other B frames are dropped, the transmitted B frame can be decoded. When B\_3 and B\_4 frames are dropped, the B\_2 frame can be decoded. However, in H.264 scalable streaming, a complicated relationship exists between B frames. When B\_3 and B\_4 frames are dropped, the B\_2 frame cannot be decoded.

**3.2. AdaptiveSTB.** Our *AdaptiveSTB* decreases the dependency among scalable layers in scalable streaming, thereby enhancing the performance of the streaming service in wireless environments. For the decoding layer in a scalable stream, the decoder needs to obtain information from other reference layers and know the dependency between scalable layers. When reference layers are lost, the decoder discards the referring layers. For enhancing the quality of a scalable streaming service, it is critical to decrease such indirect loss. Our *AdaptiveSTB* converts H.264 scalable streaming into MP4 scalable streaming before transmitting layers over the wireless network.

When whole packets are available, the layer can be decoded in the scalable stream. When one packet is lost, other data in the layer cannot be used for the decoding. Therefore, validation of the layer can be assured only when all its packets are available. Based on the terms in Table 1, the validity of the scalable layer is given by

$$VLI_{g,f,k} = \prod_{s=1}^{NP_{g,f,k}} (1 - PLR(P_{g,f,k,s})). \quad (1)$$

TABLE 2: Scalable streams.

Layer	QP	Frame rate	Frame size
0	38.0	15	320 × 240
1	32.0	30	320 × 240
2	30.0	30	320 × 240
3	28.0	30	640 × 480
4	26.0	30	640 × 480

When the error rate of the wireless link increases, most discarded scalable layers do not satisfy this equation. One frame is divided into several layers, so the reference layer is required to decode the referring scalable layer in the scalable streaming service. The number of scalable layers available is based on the vertical dependency among scalable layers. The validity of the scalable layer is given by

$$VLV_{g,f,k} = \prod_{j=1}^k \prod_{s=1}^{NP_{g,f,j}} (1 - PLR(P_{g,f,j,s})). \quad (2)$$

Finally, the decoder should check whether reference frames are available. The decoder does not require all the scalable layers of the reference frame to decode the referring frame. If the first layer of the reference frame is available, the reference frame can be decoded, and the validity of the scalable layer in the stream is given by

$$VLH_{g,f,k} = \prod_{j=1}^{NRF_g} (VLH_{g,RF(j,g,f),1}) \times \prod_{j=1}^k \prod_{s=1}^{NP_{g,f,j}} (1 - PLR(P_{g,f,j,s})). \quad (3)$$

When the error rate of the wireless link decreases, most of the discarded scalable layers will not satisfy this equation.

## 4. Experimental Result

For verifying the performance of our *AdaptiveSTB*, we conducted a network simulation on an NS-2 simulator [32] based on data extracted from real scalable streaming data. We downloaded five movie trailer clips and one video clip from the Internet, then generated scalable layers from them using a scalable encoder.

**4.1. Scalable Multimedia.** We used the Joint Scalable Video Model (JSVM) [33] for generating scalable layers from six H.264 streams. We created five scalable layers from several original streams. The following configuration is used for generating scalable streams. QP in Table 2 stands for a quantization parameter. This value divides pixel information at each frame. Therefore, detailed pixel information for each frame is saved when QP is small. The frame rates indicate how many frames are displayed in a second. High-frame-rate streams achieve smooth transitions between frames. The frame size gives the width and height of a scalable stream.

TABLE 3: Scalable streams.

Frame name	Genre	Number of frames
Amazing Caves	Adventure	2031
The Bourne Ultimatum	Action	2125
I Am legend	Drama	2397
Fantastic 4	Action	3017
Foreman	Video Clip	399
To the Limit	Adventure	919

A scalable stream with a large frame size can hold more pixel information.

Layer 0 is encoded at 15 frames per seconds (fps) with a QP of 38. In addition, the resolution of the base layer is suitable for a 320 × 240 display. When Layer 1 is added, the frame rates are increased to 30 fps and QP is decreased to 32. This provides a clear scene for the user. As more scalable layers become available at the scalable decoder, the quality of the scalable streaming service increases. Of the movie and video clips we used (see Table 3) for simulation, *Amazing Caves* and *To the Limit* are adventure movie, so scenes can change quickly. *The Bourne Ultimatum* and *Fantastic 4* are action movies where variance between frames is large. Scenes do not change quickly in *I Am Legend* and *Foreman*.

**4.2. Simulation Environments.** Figure 6 shows an overall diagram of our network simulation with scalable streams. For verifying the performance of our *AdaptiveSTB*, we generated real scalable layers using the JSVM codec from real media streams and then obtained type, time, and size of each frame for a scalable layer. Based on gathering frame information from scalable layers, we ran a network simulation using an NS-2 simulator. In the network simulation, a stream server transfers multimedia data based on the obtained size information from real scalable layers. The capacity of the wired connection between the stream server and the STB was 100 Mbps; the wireless nodes were connected through a 10 Mbps wireless link. We ran simulations with various error rates.

In Figure 6, the client for streaming service checks the arrival time of each incoming packet from the stream server. If the packet has already been transmitted at the obtained frame time of the multimedia data in the packet, the client for streaming service can decode the multimedia data in the packet. For example, multimedia data that should be displayed four minutes after starting play is delivered three minutes after the first multimedia data arrived. The delivered multimedia data can be decoded at the client for streaming service. However, if the multimedia data are delivered five minutes later, they are discarded.

MPEG standards recommend that the decoder skip corrupted multimedia data in the next synchronization position (e.g., *start code* or *resync code*) to reduce errors. The STB can check the received scalable layer for detectable corrupted scalable layers and then skip the corrupted multimedia data caused by other dropped or delayed packets. However, because we cannot use real multimedia data in our network

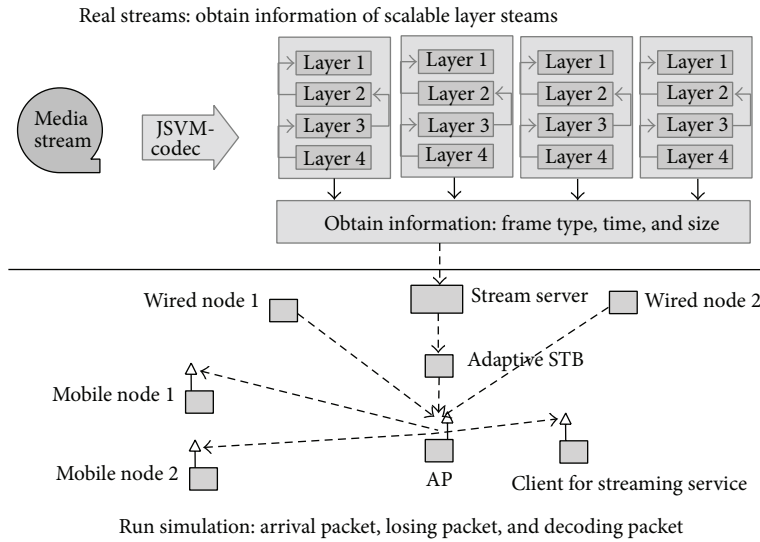


FIGURE 6: Simulation environment.

simulation, it is difficult to ascertain how much multimedia data are corrupted by lost or delayed packets.

For detecting corrupted multimedia data in the simulation, the stream server adds more information to the generating packets, including `frame_no`, `frame_seq`, `layer_id`, and `frame_flag`. Here, `frame_no` stands for the order of the transmitted frame, and `frame_seq` is the sequence number of the packets. Our *AdaptiveSTB* can detect lost packets using `frame_seq`. The label `layer_id` identifies the scalable layer as Layer 0, Layer 1, Layer 2, Layer 3, or Layer 4. Lastly, `frame_flag` indicates whether the packet is the first packet (`frame_flag = 0`), an intermediate packet (`frame_flag = 1`), or the last packet (`frame_flag = 2`) in the frame.

Figure 7 shows that each packet contains four pieces of information in the following order: `frame_no`, `frame_seq`, `frame_flag`, `layer_id`, and data. In the information, the first information shows `frame_no`. Our *AdaptiveSTB* identifies packets using `frame_no`, `frame_seq`, and `layer_id`. The decoder at the client for streaming service checks whether scalable layers are available based on additional information including `frame_no`, `frame_seq`, `layer_id`, and `frame_flag`.

### 4.3. Simulation Results

**4.3.1. Indirect Loss.** Figure 8 shows the ratio between indirect lost multimedia data and received multimedia data from the STB. In the figure, the  $x$ -axis is the error rate over the wireless network, and the  $y$ -axis is the ratio between indirect lost multimedia data and received multimedia data in the client for streaming service. The interframe encoding scheme in the MPEG standard means that some portions of the frame are referred from other frames, but this scheme increases the dependency among frames and the possibility of discarding received frames by the client for streaming service. Such a discarding of frames reduces the chance to transmit other scalable layers. In our simulation based on real scalable streams, MP4 scalable streaming outperformed

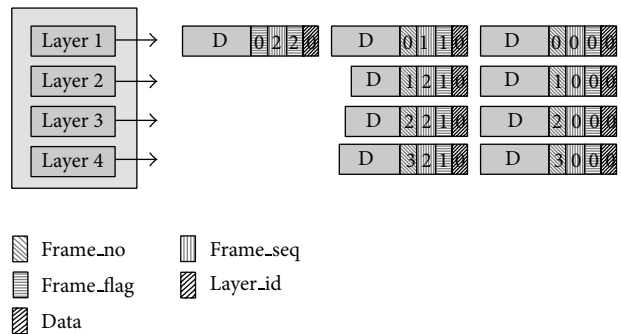


FIGURE 7: Simulation packet management.

H.264 scalable streaming. The high complexity of H.264 scalable streaming increases the number of discarded scalable layers indirectly.

*Amazing Caves* is a high-quality stream; therefore, the size of one frame is huge. It is difficult for all the packets in the frame to be delivered at the client for streaming service before decoding the frame. There is a gap between MP4 scalable streaming and H.264 scalable streaming when the low error rate over the wireless network is low, but when error rate increases, there is no difference between the two schemes. Most scalable layers do not satisfy (1), so incomplete scalable layers are discarded directly. The simulation results of *The Bourne Ultimatum* and *I Am Legend* appear to be similar to those of *Amazing Caves*. At low error rate, the ratio between indirect lost scalable layers and received scalable layers of MP4 scalable streaming is smaller than that of H.264 scalable streaming.

In *Fantastic 4*, *Foreman*, and *To the Limit*, the size of frames is relatively small. The small number of packets generated in one frame increases the possibility of decoding the scalable layer. The client for streaming service decodes scalable layers according to (3).

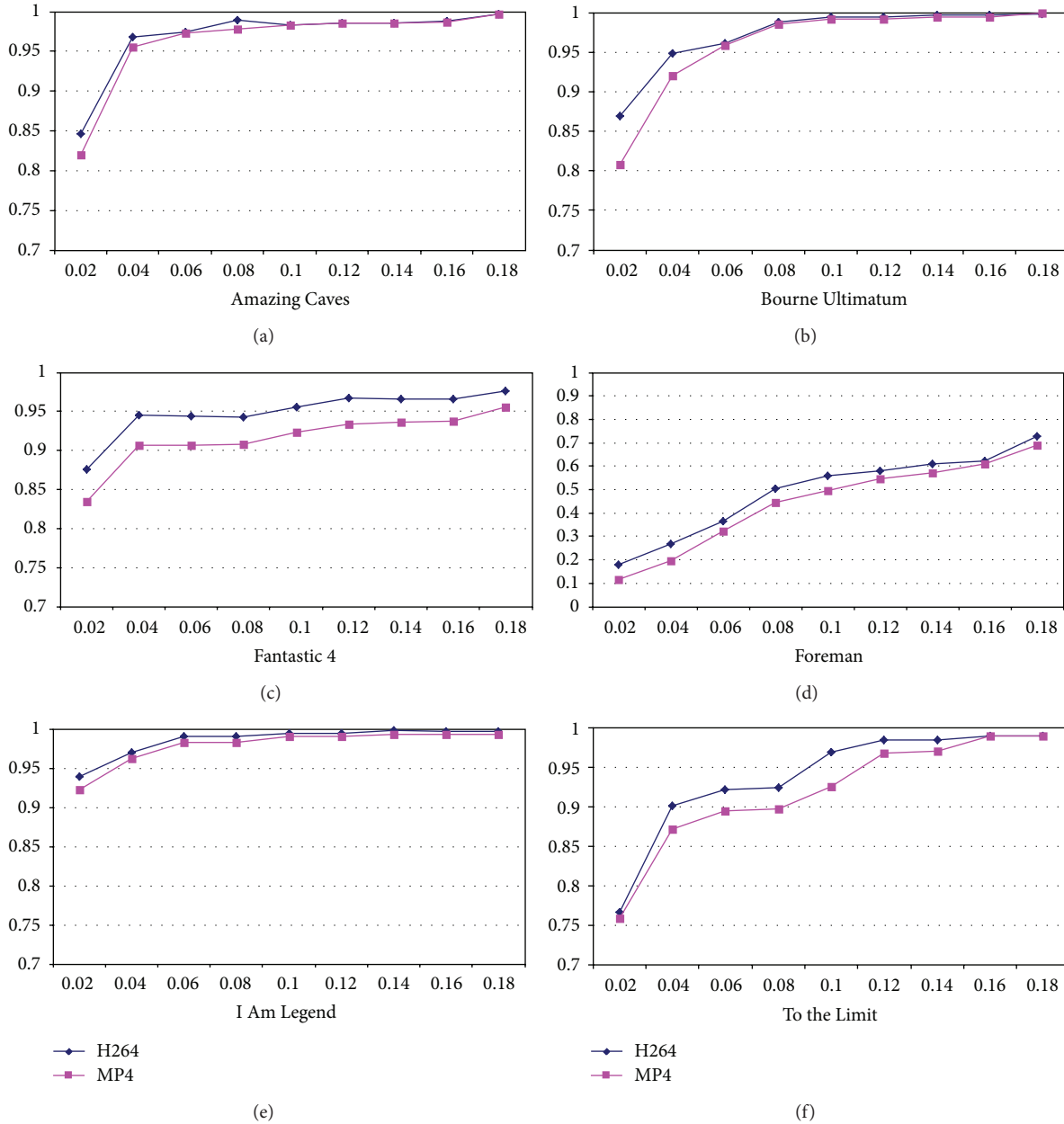


FIGURE 8: Ratio between indirect lost layers and received layers.

4.3.2. *Decoding Frame.* Figures 9 and 10 show the ratio between transmitted multimedia data and decoded multimedia data. Figure 9 shows how many bytes are displayed, and Figure 10 shows how many frames are available to the user. These two graphs show similar results. However, the ratio between B frames and I frames in the stream gives a different result.

In Figure 9, the  $x$ -axis is the error rate of the wireless network and the  $y$ -axis is the ratio between the size of the decoded multimedia data at the client for streaming service and the size of the transmitted multimedia data from the STB. High ratios mean that most of the transmitted multimedia data from the STB are decoded, so streaming services with high ratios can provide clear streams to users.

In *Amazing Caves*, MP4 scalable streaming exhibits a higher ratio than H264 scalable streaming at lower error rates, but the two schemes are similar at high error rate. Most multimedia data are dropped because they do not satisfy (1). *The Bourne Ultimatum* and *I Am Legend* exhibit similar results.

In *Fantastic 4*, *Foreman*, and *To the Limit*, MP4 scalable streaming exhibits higher ratios than H264 scalable streaming at all error rates, which means that more multimedia data are transmitted in MP4 scalable streaming. The size of frames in *Fantastic 4*, *Foreman*, and *To the Limit* are relatively smaller than those of other streams. One frame is divided into a small number of packets, so more multimedia data can satisfy (1). The multimedia data are decoded by using (3).



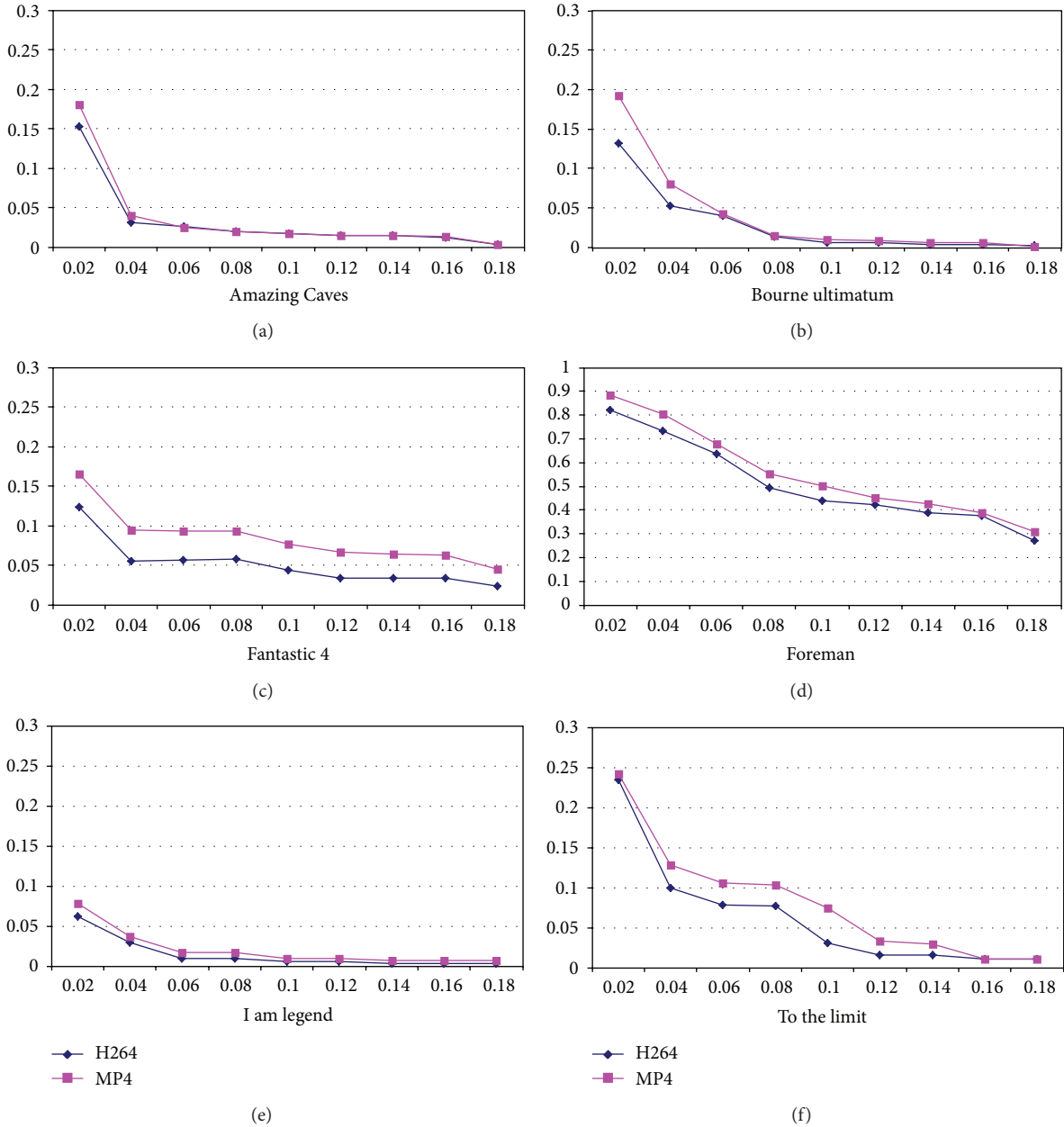


FIGURE 9: Ratio between decoded layers and sent layers (based on bytes).

Figure 10 shows the ratio between transmitted scalable layers from the STB and decoded scalable layers at the client for streaming service. The  $x$ -axis is the error rate of the wireless network and the  $y$ -axis is the ratio between decoded scalable layers at the client for streaming service and transmitted scalable layers from the STB. High ratios between decoded layers and transmitted layer indicate that many scalable layers are decoded among the transmitted scalable layers, meaning that the STB provides a good quality scalable streaming service.

In the *Amazing Caves*, a gap between MP4 scalable streaming and H264 scalable streaming is distinguishable at low error rate. However, the gap closes at high error rate

as the result of the discarding of most scalable layers. Such simulation results follow according to (1). In *The Bourne Ultimatum* and *I Am Legend*, more frames in MP4 scalable streaming are decoded than in H264 scalable streaming at low error rate, but, as error rate increases, the client for streaming service drops more receiving scalable layers following (1). In *The Bourne Ultimatum*, H264 scalable streaming is better than MP4 scalable streaming even at some high error rates. At low error rate for *The Bourne Ultimatum*, the ratio of scalable layers in Figure 10 is higher than the ratio of multimedia data in Figure 9. *The Bourne Ultimatum* contains a high ratio of B frames, so the ratio of scalable layers is increased for a small ratio of multimedia data.

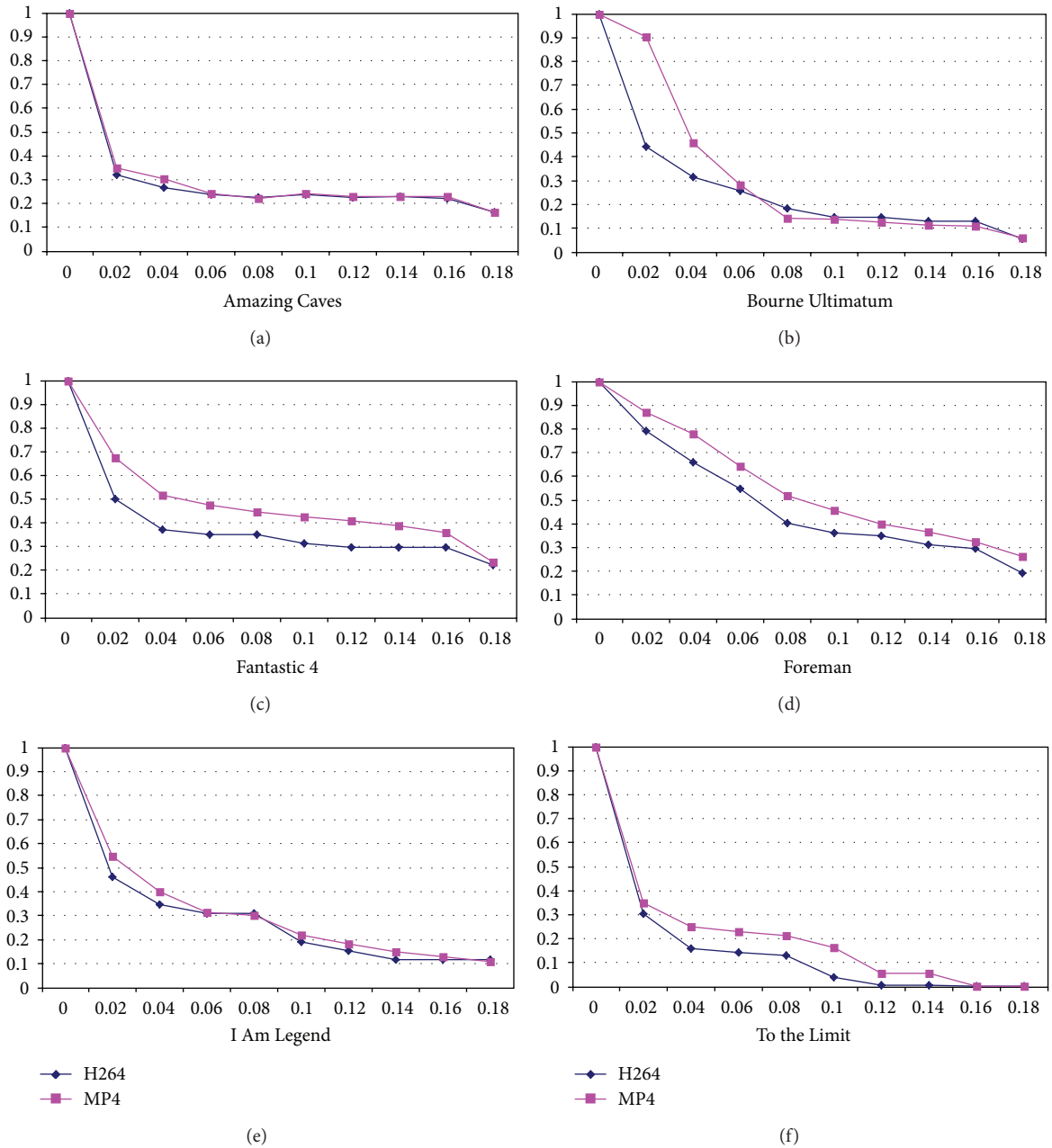


FIGURE 10: Ratio between decoded layers and sent layers (based on frames).

In *Fantastic 4*, *Foreman*, and *To the Limit*, MP4 scalable streaming outperforms H264 scalable streaming for all error rates. In those streams, one frame is divided into a small number of packets, so the scalable layers themselves are available from (1). Most of discarded scalable layers do not satisfy (3).

Our simulation results show that a scalable scheme with low dependence among scalable layers provides good service to users. At low wireless network error rates, the relationship among scalable layers determines the quality of scalable streams; this is especially critical for streams with small sized frames.

### 5. Conclusion

In this paper, we show that our *AdaptiveSTB* converts scalable layers with complicated dependency into simple scalable layers, thereby enhancing the scalable streaming service over a wireless network. We found that the main reason for a reduction in quality of scalable streaming over a wireless network had to do with the error rate. When a scalable layer can only be delivered with a high error rate, the dependency among scalable layers exerts little influence on the quality of the scalable streaming service. However, when the error rate of the wireless network is low or the size of scalable

layers is small, the quality of the scalable streaming service is determined by the dependency among scalable layers.

We perform packet-level analysis for scalable streaming service over a wireless network. Additionally, we suggest formulas for the expected quality of the scalable streaming service and prove the performance of our *AdaptiveSTB* through simulations in wireless networks. We compare the performance of scalable streams over wireless networks with various error rates. Future work will address limitation of resources (e.g., memory and computing power) at the set-top box as well as various network environments.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This paper is a revised and extended version of a paper that was originally presented at the 2014 FTRA International Symposium on Frontier and Innovation in Future Computing and Communications. This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) and funded by the Ministry of Education (2013R1A1A2063006).

## References

- [1] "Cisco-Visual-Networking-Index," Cisco Systems, 2010–2015, <http://bit.ly/pIDtBX>.
- [2] J. Erman, A. Gerber, K. K. Ramadrihnan, S. Sen, and O. Spatscheck, "Over the top video: the gorilla in cellular networks," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference (IMC '11)*, pp. 127–136, New York, NY, USA, November 2011.
- [3] J. C. Tsai and N. Y. Yen, "Cloud-empowered multimedia service: an automatic video storytelling tool," *Journal of Convergence*, vol. 4, no. 3, pp. 13–19, 2013.
- [4] O. Mirzaei and R. M. Akbarzadeh-T, "A novel learning algorithm based on a multi-agent structure for solving multi-mode resource-constrained project scheduling problem," *Journal of Convergence*, vol. 4, no. 1, pp. 47–52, 2013.
- [5] V. V. M. Nhat and N. H. Quoc, "A model of adaptive grouping scheduling in obs core nodes," *Journal of Convergence*, vol. 5, no. 1, pp. 9–13, 2014.
- [6] A. K. Gopalakrishnan, "A subjective job scheduler based on a backpropagation neural network," *Human-Centric Computing and Information Sciences*, vol. 3, article 17, 2013.
- [7] H. Y. Hsueh, C. N. Chen, and K. F. Huang, "Generating meta-data from web documents: a systematic approach," *Human-Centric Computing and Information Sciences*, vol. 3, article 7, 2013.
- [8] N. Ibrahim, M. Mohammad, and V. Alagar, "Publishing and discovering context-dependent services," *Human-Centric Computing and Information Sciences*, vol. 3, no. 1, 2013.
- [9] H. Bai, A. Wang, Y. Zhao, J. S. Pan, and A. D. Abraham, *Multiple Description Coding: Principles, Algorithms and Systems*, Springer, New York, NY, USA, 2011.
- [10] T. Gan, L. Gan, and K.-K. Ma, "Reducing video-quality fluctuations for streaming scalable video using unequal error protection, retransmission, and interleaving," *IEEE Transactions on Image Processing*, vol. 15, no. 4, pp. 819–832, 2006.
- [11] K. Park and W. Wang, "Afec: an adaptive forward error-correction protocol for end-to-end transport of real-time traffic," in *Proceedings of the IEEE International Conference on Computer Communication and Networks*, pp. 196–205, Lafayette, La, USA, 1997.
- [12] S. Jakubczak and D. Katabi, "A cross-layer design for scalable mobile video," in *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking (MobiCom '11)*, pp. 289–300, September 2011.
- [13] S. Aditya and S. Katti, "FlexCast: graceful wireless video streaming," in *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking (MobiCom '11)*, pp. 277–288, September 2011.
- [14] J. J. Quinlan, A. H. Zahran, and C. J. Sreenan, "ALD: adaptive layer distribution for scalable video," in *Proceedings of the 4th ACM Multimedia Systems Conference (MMSys '13)*, pp. 202–213, March 2013.
- [15] Z. Li, Y. Huang, G. Liu, F. Wang, Z.-L. Zhang, and Y. Dai, "Cloud transcoder: bridging the format and resolution gap between Internet videos and mobile devices," in *Proceedings of the 22nd ACM Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV '12)*, pp. 33–38, June 2012.
- [16] A. G. Kunzel, H. Kalva, and B. Furht, "A study of transcoding on cloud environments for video content delivery," in *Proceedings of the ACM Workshop on Mobile Cloud Media Computing (MCMC '10)*, pp. 13–18, October 2010.
- [17] Z. Huang, C. Mei, L. E. Li, and T. Woo, "CloudStream: delivering high-quality streaming videos through a cloud-based SVC proxy," in *Proceedings of the INFOCOM 2011*, pp. 201–205, April 2011.
- [18] S. Ko, S. Park, and H. Han, "Design analysis for real-time video transcoding on cloud systems," in *Proceedings of the 28th Annual ACM Symposium on Applied Computing (SAC '13)*, pp. 1610–1615, March 2013.
- [19] B. Shen, S. Lee, and S. Basu, "Caching strategies in transcoding-enabled proxy systems for streaming media distribution networks," *IEEE Transactions on Multimedia*, vol. 6, no. 2, pp. 375–386, 2004.
- [20] S.-P. Chan, C.-W. Kok, and A. K. Wong, "Multimedia streaming gateway with jitter detection," in *Proceedings of the International Conference on Communications (ICC '03)*, pp. 1875–1879, May 2003.
- [21] C.-M. Huang, T.-H. Hsu, and C.-K. Chang, "A proxy-based adaptive flow control scheme for media streaming," in *Proceedings of the ACM Symposium on Applied Computing*, pp. 750–754, New York, NY, USA, March 2002.
- [22] L. Gao, Z.-L. Zhang, and D. Towsley, "Proxy-assisted techniques for delivering continuous multimedia streams," *IEEE/ACM Transactions on Networking*, vol. 11, no. 6, pp. 884–894, 2003.
- [23] P. Schojer, L. Böszörményi, H. Hellwagner, B. Penz, and S. Podlipnig, "Architecture of a quality based intelligent proxy (QBIX) for MPEG-4 videos," in *Proceedings of the 12th International Conference on World Wide Web (WWW '03)*, pp. 394–402, Budapest, Hungary, May 2003.
- [24] S.-P. Chan, C.-W. Kok, and A. K. Wong, "Multimedia streaming gateway with jitter detection," *IEEE Transactions on Multimedia*, vol. 7, no. 3, pp. 585–592, 2005.

- [25] A. Laursen, J. Olkin, and M. Porter, "Oracle media server: providing consumer based interactive access to multimedia data," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, vol. 23, pp. 470–477, May 1994.
- [26] S. Robertson and R. Rivin, *Analog Devices: Designing IPTV Set-Top Boxes Without Getting Boxed*, Analog Devices.
- [27] Telchemy, *Analog Devices: Designing IPTV Set-Top Boxes Without Getting Boxed*, Telchemy.
- [28] S. Acharya and B. Smith, "Middleman: a video caching proxy server," in *Proceedings of NOSSDAV*, 2000.
- [29] K. L. Wu, P. S. Yu, and J. L. Wolf, "Segment-based proxy caching of multimedia streams," in *Proceedings of the 10th international conference on World Wide*, pp. 36–44, New York, NY, USA, 2001.
- [30] O. Verscheure, C. Venkatramani, P. Frossard, and L. Amini, "Joint server scheduling and proxy caching for video delivery," *Computer Communications*, vol. 25, no. 4, pp. 413–423, 2002.
- [31] J. Liu and B. O. Li, "A QoS-based joint scheduling and caching algorithm for multimedia objects," *World Wide Web*, vol. 7, no. 3, pp. 281–296, 2004.
- [32] "Ns-2 network simulator," 2000, <http://www.isi.edu/nsnam/ns/>.
- [33] ISO/IEC JTC 1/SC29/WG 11 and ITU-T SG16 Q.6, "Joint Scalable Video Model (JSVM) Software Manual," Included in JVT-X203. ISO/IEC JTC 1/SC29/WG 11 and ITU-T SG16 Q. 6., 2007.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

