Adaptive Error Control for Scalable Video Streaming over Wireless Internet

Jen-Wen Ding[1*]   Wei-Jen Chen[1]   Chu-Fu Wang[2]

[1]Department of Information Management, National Kaohsiung University of Applied Sciences
[2]Department of Computer Science, National PingTung University of Education
[*]E-mail: jwding@cc.kuas.edu.tw

Abstract

With the wide deployment of broadband wireless networks around the world, there is a growing interest in wireless personalized video streaming. However, the bit error rate (BER) of wireless networks is much higher than that of wired networks. The high BER, unfortunately, will significantly degrade the visual quality of video streaming. In this paper, we propose an adaptive error control scheme for broadcast-based video-on-demand services. The proposed scheme can adaptively allocate proper forward error correction codes to different video streams according to the measured network conditions, such that the average visual quality of all clients can be maximized.

Keywords: Broadband wireless networks, video streaming, adaptive error control scheme, scalable video coding

1. Introduction

With the rapid deployment of broadband wireless networks around the world, there is a growing interest in wireless personalized video streaming. It can be anticipated that many service providers will create numerous interesting wireless multimedia applications that make use of video streaming in the near future. However, it is difficult to provide smooth real-time video streams over wireless networks. One of the main problems is that the BER of wireless networks (usually on the order of 10^{-3} and sometimes as high as 10^{-1}) is usually much higher than that of wired networks (usually on the order of 10^{-6} to 10^{-8}) [1]. The high BER is caused by many factors, such as fast fading, slow fading, co-channel interference, adjacent channel interference, and noise disturbances. Unfortunately, the high BER will significantly degrade the visual quality of video streaming. This paper focuses on dealing with negative impact caused by the high wireless BER on the visual quality of video streaming.

In the following, we briefly review recent previous work on the error control for video streaming. Error control for video streaming can be classified into four categories: error resilience, error concealment, FEC, and ARQ (automatic repeat request). The first type of approach involves the design of smart encoders, which attempt to limit the scope of the visual damage caused by lost data. The second category involves the design of smart decoders, which attempt to hide the lost data using received data. The third and four categories involve encoding and retransmitting redundant data. In the past few years, the error control for scalable video coding (SVC) has attracted much attention. Many studies dealing with the error control for layered multicast video streams have been proposed [2]. In addition to SVC, the error control for streaming video over wired/wireless hybrid networks also attracts much attention [3]. Many studies have been devoted to this research issue.

Although much research have been done on the error control for unicast and multicast video streaming, little research has been done on the error control for broadcast-based on-demand video streaming (i.e., near video-on-demand services). In practice, because the available bit rate of wireless networks is usually very limited and must be shared by a large number of users simultaneously, it can be expected that broadcast-based video-on-demand (VoD) will play an important role in the future wireless video streaming. In this paper, we proposed a quality-aware error control scheme for broadcast-based VoD streaming, called adaptive FEC allocation (AFA). Unlike previous work, AFA takes into account the special property of VoD broadcasting protocols in FEC allocation. Our simulation results show that AFA can effectively improve the perceptual quality of video streaming over wireless internet.

The rest of this paper is organized as follows. Section 2 describes the system architecture. Section 3 presents the main idea behind AFA and gives the C-like pseudocode of AFA. Section 4 shows the simulation results, and Section 5 concludes this paper.
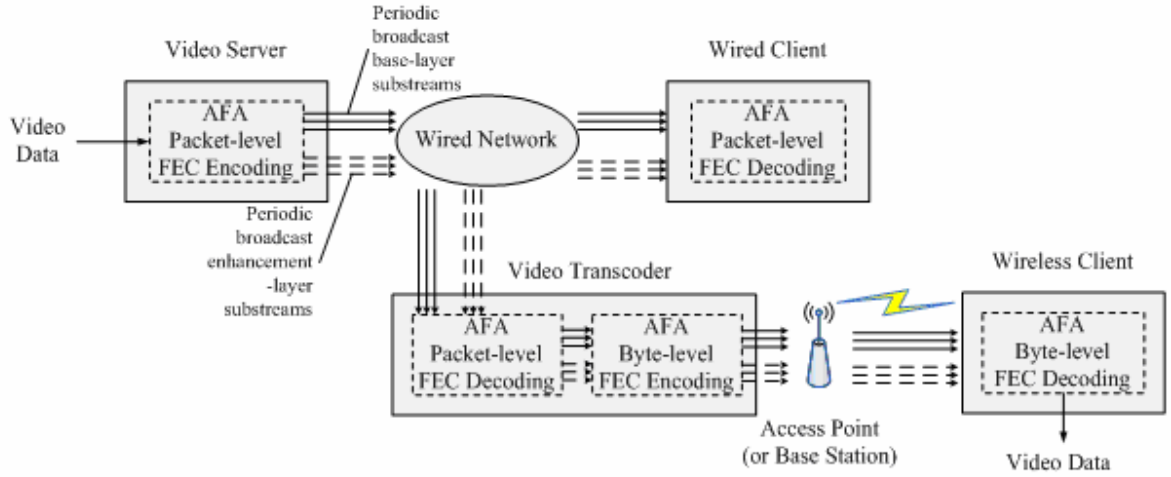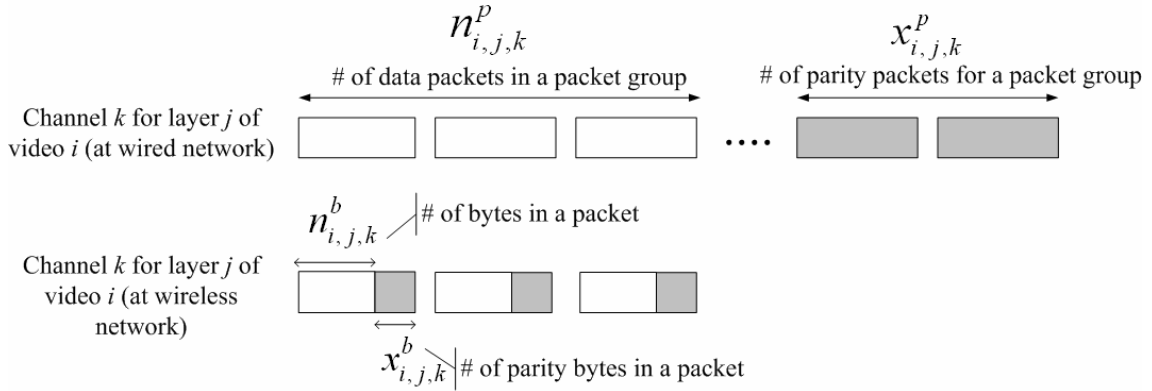
Fig. 1: System Architecture.

$$n^p_{i,j,k}$$
# of data packets in a packet group

$$x^p_{i,j,k}$$
# of parity packets for a packet group

Channel k for layer j of video i (at wired network)

$$n^b_{i,j,k}$$ # of bytes in a packet

Channel k for layer j of video i (at wireless network)

$$x^b_{i,j,k}$$ # of parity bytes in a packet

Fig. 2: Relation between key parameters.

2. System Architecture

Because of the real-time nature of video streaming, instead of ARQ, we employ FEC to perform error recovery. Previous work has indicated that in wired networks, packets are dropped mainly due to congestion at routers, while in wireless networks, packets are often lost due to the high BER. In view of this, we employ packet-level FEC to recover the packet loss in wired networks and byte-level FEC to recover the packet loss in wireless networks. Fig. 2 shows our system architecture. The architecture discussed in this paper is similar to that proposed by T.-W. Lee et al. in [3]. There are two main differences between our work and their work. First, our work considers periodic broadcast video streams while their work considers multicast video streams. Specifically, in our architecture, multiple logical channels are used to periodically broadcast a video, while in their architecture, only a logical channel is used to multicast

a video. Second, we assume that the data rate of each layer of a video is predetermined while they assume that the data rate of each layer of a video can be adjusted in real time. We make this assumption because the data rate of most pre-stored videos is not changeable.

In this paper, we employ scalable video coding, i.e., a video stream is encoded into one base-layer substream and multiple enhancement-layer substreams. We adopt fast broadcasting (FB) as the VoD broadcasting protocol for its simplicity in implementation. FB divides a video file into equal-sized segments and then broadcast these segments with different frequencies. For more details of FB, refers are referred to [4]. As shown in Fig. 2, at the video server, packet-level FEC are encoded for each broadcast video stream. Packet-level FEC are in the form of parity packets and can be used to recover the packet loss caused by network congestion. For wired clients, the error correction is carried out directly

at the client sides using the parity packets. For wireless clients, the video streams will pass a video transcoder before reaching the clients. The video transcoder performs the following two tasks: it (1) uses the received parity packets to recover the video packets lost during the wired networks, and (2) encodes byte-level FEC for each broadcast video stream before broadcasting them to the wireless clients. Byte-level FEC are in the forms of parity bytes and can thus be used to recover bit errors.

The two AFA modules, packet-level AFA module and byte-level AFA module, can adaptively adjust the number of parity packets and parity bytes to be allocated to each broadcast video stream according to the measured network conditions.

3. Adaptive FEC Allocation Scheme

For the sake of space, we discuss only byte-level FEC allocation in this paper. The algorithm of packet-level FEC allocation is similar to that of byte-level FEC allocation. Table 1 lists the notation used in this paper. Fig. 2 shows the relation between the key parameters. In the following, we briefly explain the main idea behind the byte-level AFA scheme.

Table 1. Summary of Notation

V	Number of videos
L_i	Number of layers of video i
ζ_i	Access probability of video i
C_i	Number of broadcast channels for each video layer of video i
$r_{i,j}$	Data rate of layer j of video i
B_{wd}	Available bandwidth in the wired network
B_{ws}	Available bandwidth in the wireless network
κ_α	Estimated maximum packet loss rate in the wired network using interval estimate with confidence of α
ε_α	Estimated maximum bit error rate in the wireless network using interval estimate with confidence of α
τ_b	Maximum byte error rate in the wireless network
m	Number of bits of a byte
$\delta_{i,j}^{max}$	Permissible packet loss rate for layer j of video i
$\delta_{i,j,k}^{p}$	Packet loss rate for channel k of layer j of video i by applying packet-level FEC
$\delta_{i,j,k}^{b}$	Packet loss rate for channel k of layer j of video i by applying packet-level FEC
$n_{i,j,k}^{p}$	Number of packets of a packet group for channel k of layer j of video i
$n_{i,j,k}^{b}$	Number of bytes of a packet for channel k of

	layer j of video i
$x_{i,j,k}^{p}$	Number of parity packets in a packet group for channel k of layer j of video i
$x_{i,j,k}^{b}$	Number of parity bytes in a packet for channel k of layer j of video i

Since the network conditions (such as available bit rate, delay, and loss rate) of both wired and wireless networks vary with time, AFA scheme is carried out periodically according to the latest network conditions. We assume that AFA is executed once in the delivery of each video segment. In practice, the period of the AFA execution time is a changeable parameter, which depends on the fluctuation rate of the network conditions. In each period, AFA first sorts all broadcast video streams according to their visual importance, and then allocates FECs to these streams accordingly. Since fast broadcasting algorithm employs equal-sized video segments, the visual importance of each video segment is dominated by the number of clients receiving the segment. It can be observed that the number of clients is proportional to (1) the access probability of the video to which the segment belongs and (2) the inverse of the broadcasting frequency of the segment. In view of this, we sort all video streams according the multiple of the two values. Fig. 3 shows the C-like pseudocode for sorting all video streams.

Byte-level AFA performs as follows. It first considers the base layer of all video streams, and then the first enhancement layer of all video streams, and so on. According to the visual importance of each video stream, AFA allocates sufficient byte-level FECs to all video streams according to their visual importance. An interesting question arises as to how many parity bytes in a packet are required for a video stream. This depends on how sensitivity the video decoder in use is vulnerable to packet losses. We assume that the permissible packet loss rate of layer j of video i is $\delta_{i,j}^{max}$. Then, according to this value, we can determine the required parity bytes. Specifically, we can predict the packet loss rate of any stream, say k, that belongs to layer j of video i for a given number of parity bytes, say x, as follows:

$$\delta_{i,j,k}^{b}(x) = \sum_{g=x+1}^{n_{i,j,k}^{b}} \binom{n_{i,j,k}^{b}}{g} (\tau_b)^g (1-\tau_b)^{n_{i,j,k}^{b}-g}$$

To save the bandwidth consumed by FEC, AFA employs statistical resource allocation. Specifically, instead of considering the worst cast of network conditions, AFA employs interval estimate to find out the actual packet loss rate of both wired and wireless networks. The confidence for the interval estimate can be set to a high value, such as 99%, to avoid

underestimate the actual packet loss rate. Fig. 4 shows the C-like pseudocode for byte-level AFA.

```
// sort all video substremas according to their visual importance
 for (i = 1; i ≤ V; i + +)
    for (k = 1; k ≤ C_i; k + +)

       w_{i,k} = ζ_i × 1/f_{i,k} ;

 for (priority = 1; priority ≤ V × C_i; priority + +)  {
    find {i, k | w_{i,k} ≥ w_{x,y}, x ≠ i, y ≠ k};
    vid[priority] = i;
    cid[priority] = k;
}
```

Fig. 3: Sorting all video streams according to their visual importance.

```
// phase II: allocate byte-level FECs among all video streams
estimate ε_α;
τ_b = 1 − (1 − ε_α)^m;
for (j = 1; j ≤ L_i;  j + +)  {
  priority = 1;
  for (priority = 1;  priority ≤ V × C_i;  priority + +)  {
    i = vid[priority];
    k = cid[priority];
    set x^b_{i,j,k} = min{x | δ^b_{i,j,k}(x) ≤ δ^max_{i,j}}

       where δ^b_{i,j,k}(x) = Σ_{g=x+1}^{n^b_{i,j,k}} ( n^b_{i,j,k} / g )(τ_b)^g (1 − τ_b)^{n^b_{i,j,k} − g} ;

    set x^b_{i,j,k} = 2 × x^b_{i,j,k} ;

    if (B_{ws} ≥ x^b_{i,j,k}/n^b_{i,j,k} × r_{i,j})

       B_{ws} = B_{ws} − x^b_{i,j,k}/n^b_{i,j,k} × r_{i,j};

    else  {
         set x = B_{ws};
         B_{ws} = 0;
    }
    priority  =  priority + 1;
  }
}
```

Fig 4: Byte-level AFA scheme.

4. Performance Evaluation

To verify the effectiveness of the proposed AFA scheme, we conducted a series of simulations under different network conditions. The main simulation parameters are described as follows. The distribution of wireless BER is assumed to be a normal distribution. The mean is set to 10^{-1}, 10^{-2}, and 10^{-3}, respectively. The standard deviation is set to a tenth of the mean. The confidence of the interval estimate for packet loss is set to 99%. Ten videos were tested. Each video consists of two layers: one base layer and one enhancement layer. The bit rate of each layer is assumed to be 300kbps. Five channels are used to broadcast each layer of a video. The average PSNRs of the base layer and the enhancement layer is assumed to be 32 and 8, respectively. The permissible packet loss rate of each layer is set to 1%. Three different FEC allocation schemes are compared, AFA, UFA, and PFA. UFA treats all video streams equally in FEC allocation. PFA is similar to AFA, but it determines the visual importance of video streams according to only the access probability of the videos. Fig. 5 shows the performance comparison result.
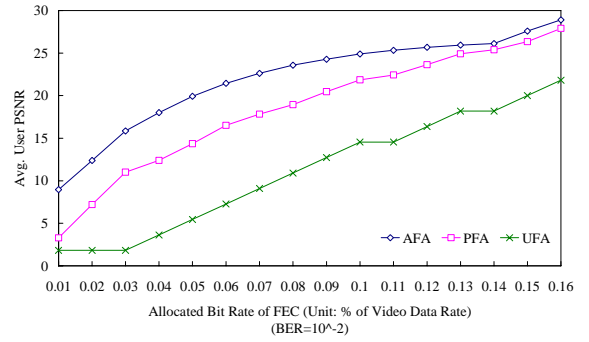

Fig. 5: Avg. user PSNR (BER=10^{-2})

5. Conclusions

In this paper, we propose a novel error control scheme, AFA, for scalable video broadcasting over wireless Internet. AFA employs both packet-level FEC and byte-level FEC to recover the packet losses in wired and wireless networks. Our simulation results show that AFA can achieve a high error recovery rate using a small amount of FEC control information.

6. References

[1] W. C. Y. Lee, Mobile Communications Design Fundamentals, 2nd Edition, John Wiley and Sons, 1993.

[2] W.-T. Tan et al., "Video multicast using layered FEC and scalable compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, 2001.

[3] T.-W. Angus Lee et al., "Allocation of layer bandwidths and FECs for video multicast over wired and wireless networks," *IEEE Trans. Circuits Syst. Video Technol*, Vol. 12, No. 12, 2002.

[4] L.-S. Juhn et al., "Fast data broadcasting and receiving scheme for popular video service," *IEEE Trans. Broadcast.*, vol. 44, no.1, pp.100-105, 1998.