

# PRIORITIZED RETRANSMISSION FOR ERROR PROTECTION OF VIDEO STREAMING OVER WLANS

*Hsiao-Cheng Wei, Yuh-Chou Tsai, and Chia-Wen Lin*

Department of Computer Science and Information Engineering,  
National Chung Cheng University, Chiayi, Taiwan, R.O.C.  
E-mail: cwlin@cs.ccu.edu.tw

## ABSTRACT

Video transport over Wireless LANs (WLANs) may suffer from signal fading, noise interference, and network congestion which will cause packet loss or packet error. Should a packet loss occur in some frame, it will not only affect the corresponding frame but may also lead to error propagation to the following frames until reaching the next intra-coded frame. In addition, the bandwidth of WLAN is usually far less than that of wired networks, thus protection mechanisms for error control has an innate limitation. In this paper, we first propose a scheme to estimate the error propagation effect of a concealed lost packet. Consider the error-prone characteristic of WLANs and the scenario under limited resource; we propose a prioritized retransmission mechanism to protect against the bursty packet losses in WLAN environments. Experimental results show that the proposed method can usually significantly improve the quality of video streaming over WLANs as compared to play-out deadline-based methods.

## 1. INTRODUCTION

Due to the simplicity of configuration and the relatively low cost for network setup, accessing the Internet via WLANs becomes more and more popular. Using mobile devices, notebooks, or PDAs, for example, WLAN is available for various applications anywhere in campuses, in offices, and even at home. Among all applications, multimedia applications, such as multimedia streaming, multimedia messaging, video telephony, and video-on-demand are the most interesting and popular applications, since multimedia applications are easier to be accepted, and are closer to human life. WLANs play a key role in home networks. Fig. 1 shows a three-tier streaming system which can be divided into two parts, the first including the path from the ISP streaming server to the home server(s) and the second containing the path from the home server, which connects to the wireless access point (AP), to wireless terminals. This three-tier architecture is getting more and more popular in home media networking applications. However, several characteristics of WLANs, such as limited bandwidth, high data error/loss rate, and unstable network condition, pose a great challenge on enabling multimedia applications. High data error/loss rate, bursty packet loss, and heterogeneous channel conditions pose great impact on video applications. These threats result from signal fading, interference, noise, congestion, or the hidden

terminal problem. The bit errors in a packet may cause the loss of a whole packet, if the number of corrupted bits goes beyond the error correction capacity of error correction codes. Packet loss may lead to serious video quality degradation or even make the video clip un-decodable.

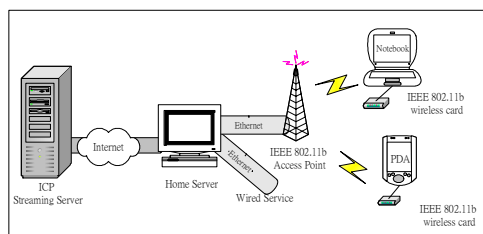


Fig. 1. A three-tier video streaming system.

Error protection for video streaming over WLAN has been an active research topic [1-5]. Automatic Retransmission reQuest (ARQ) is one of the most commonly used channel coding schemes for error protection, which is particularly useful to combat against burst errors. For example, a typical scheduling-oriented retransmission method was presented in [1] which takes into account the play-out deadline based on the Early-Deadline-First (EDF) principle. The EDF method chooses to retransmit the lost packets with earliest play-out deadlines, which may lead to the overriding of regular packets by the retransmitted packets. In [2], the authors proposed a class of packet scheduling algorithms for wireless video streaming by applying different deadline thresholds to video packets of different importance. In [3], five schemes based on Uniform, Frame Level Reference, Slice Level Reference, Motion, and Motion + Slice Level Reference, are respectively proposed for loss differentiation. In [4] a novel packet-level multiple-description forward error correction (MD-FEC) was proposed to resolve the heterogeneity of client channel conditions for multicasting over WLANs. Because each video packet contributes different importance to the video, retransmitting loss packets should be scheduled by its importance. In this paper, we propose a prioritized ARQ scheme for channel transcoding in the media gateway (e.g., a home server) to enhance the error robustness of streaming video. We consider the application scenario that the home server does not reserve dedicated bandwidth for supporting the packet retransmissions due to limited bandwidth. This implies that the retransmitted packets will compete for the limited bandwidth resources with the regular video packets. The proposed priority-based retransmission scheme is adopted with packet importance information. The level of packet importance is measured by

estimating the error propagation effect caused by the corresponding packet loss. In our method, this measurement is performed only once and the results will be stored at the streaming server for guiding the packet retransmission scheduling and decision.

The rest of this paper is organized as follows. In Section 2, we present our proposed adaptive ARQ scheme. In Section 3, experiments are conducted to justify the feasibility of our methods. Finally, conclusion is drawn in Section 4.

## 2. PROPOSED PRIORITIZED RETRANSMISSION SCHEME

In this work, we adopt the concept of application-level framing (ALF) defined in [10] for packetization, in which packets with a corrupted bit-number greater than the capacity of error correction will be dropped. In this way, the WLAN channel becomes a packet erasure channel. As shown in Fig. 2, when a video packet in a frame (e.g., an I-frame in this example) is not correctly received by the client, the error propagation will affect the following MBs of subsequent frames, which reference to the lost MBs. The goal of this work is to estimate how much distortion a lost packet will cause after performing error concealment, here we assume the zero-motion error concealment [6] is adopted in the video decoder. The results of error propagation estimation are stored at the streaming server for guiding the packet retransmission scheduling and decision.

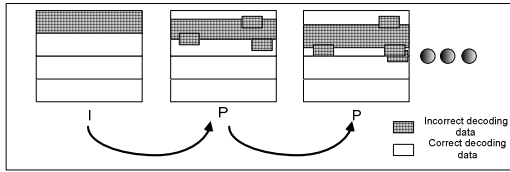


Fig. 2. Effect of error propagation due to a packet loss.

### 2.1. Estimating the impact factor of a lost packet

In our work, only the impact values of the lost packets belonging to I- and P-frames need to be estimated, since the packet loss within a B frame won't result in any error propagation. Besides, the estimation is performed within a GOP, because the error propagation is constrained in a GOP.

#### **Algorithm 1: Error Propagation Estimation of a Lost Packet**

for each GOP in the video sequence

  for  $m = 1$  to  $N_{\text{SLICE}}$

    for  $n = 1$  to  $N_{\text{GOP}}$

      {  
      Step 1: if  $p_n(i, j) \in \text{slice}_n^m$  then  $p_n(i, j) := \text{Loss\_Mark}$ ,

      Step 2: decode the corrupted GOP, and

      if  $p_n(i, j) = \text{Loss\_Mark}$  perform zero-motion error concealment

      Step 3: calculate the PSNR of all frames in the GOP

      Step 4: calculate the loss impact factor

$$\text{Impact}_n^m = \sum_{j=n}^{N_{\text{GOP}}} (PSNR_j - PSNR_j^{\text{EC}}) \quad (1)$$

      }

where  $N_{\text{SLICE}}$  is the number of slices per frame;  $N_{\text{GOP}}$  is the GOP size;  $\text{Slice}_n^m$  represents the  $m$ -th slice (or packet) of the  $n$ -th frame, and  $p_n(i, j)$  is the pixel  $(i, j)$  of  $n$ -th frame; Setting  $p_n(i, j) = \text{Loss\_Mark}$  is to simulate a loss on pixels belonging to  $\text{Slice}_n^m$  so that decoder will perform error concealment on the error regions accordingly. The main task of Step 1 is to locate the corrupted area of a lost packet. In Step 2, the decoder conceals the corrupted area by using zero-motion error concealment. In Step 3, we calculate the PSNR of all frames in the same GOP.  $\text{Impact}_n^m$  is the impact value of  $\text{Slice}_n^m$ , which is calculated by summing the PSNR difference between error-free frames ( $PSNR_j$ ) and the ones ( $PSNR_j^{\text{EC}}$ ) concealed from the loss of  $\text{Slice}_n^m$  as defined in Eq. (1).

For example, the plot of packet-loss impact values for the 300-frame "Foreman" sequence is shown in Fig. 3. Each video frame is encapsulated into four packets. The vertical bars from packet #1 to packet #1200 indicate the loss-impact values when the corresponding packet is lost individually. We can observe from Fig. 3 that, in general, the degree of loss-impact of packets in frames closer to the I-frame in the same GOP is usually higher. However, it is not absolutely true for all packets, which is dependent on the number of intra-coding MBs in a packet and also on the frequency of a packet being referenced by the following frames. Note, the error propagation estimation can be performed in off-line encoding for prestored video streaming application. Thus, there is no additional complexity with the proposed method, making it suitable for realtime transcoding applications.

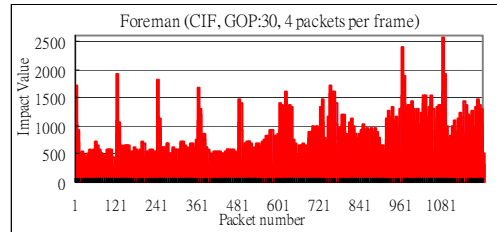


Fig. 3. Packet impact values for the "Foreman" sequence.

### 2.2. Priority-based retransmission scheme

Based on the estimated loss-impact values, we propose a prioritized retransmission scheme. Note that, the retransmitted packets will consume part of available bandwidth as well as cause extra transmission delay. In this work, we consider the application scenario that, due to limited channel bandwidth, the home server does not reserve dedicated bandwidth for supporting the lost-packet retransmissions. That means, the retransmissions of lost packets from the server may cause the resource contention between the regular video packets and the protection information. Under this constrained scenario, if an important packet gets lost but cannot be resent due to limited bandwidth, serious quality degradation may be introduced. One feasible solution to this problem is to drop some "unimportant" regular packets so as to use the saved bandwidth to retransmit the "important" lost packets under the delay constraint. The degree of importance of a packet is measured by its loss-impact value defined above.

Since the packet-loss status of communication channel is difficult to predict in advance, and what “regular” packets the server is sending while receiving a retransmission request (i.e., a NCK packet) is also dependent on the arrival time of the request, and the play-out deadline left for a retransmitted packet is also varying, it would be difficult to determine an optimal retransmission policy in real-time. We propose to use the Greedy algorithm [8] for making the retransmission decisions. In our method, the client will initially determine whether or not to request a retransmission for a lost packet according to its play-out deadline. Should the server receive a retransmission request for a lost packet, it will use the Greedy algorithm, in terms of the rank of the packet's loss-impact value, to choose the one with the larger loss-impact value to transmit from either of this lost packet and the regular packet(s) with a similar total size, and drop the other. The proposed algorithm is listed as follows:

**Algorithm 2: Priority-Based Retransmission**

**Client side:**

```

if  $(T_{\text{cur}} + \overline{RTT}_i + D_s) \geq T_d(P_i)$ 
  do not request retransmission
else
  request retransmission
  -----

```

**Server side:**

```

if receiving a retransmission request for packet  $P_i^{\text{lost}}$ 
{
  find  $P_j$  with smallest impact factors in the regular packets
  with  $\text{size}(P_j) \geq \text{size}(P_i^{\text{lost}})$  under  $T_d(P_i^{\text{lost}})$  constraint
  if  $P_j$  exists and the impact value of  $P_i^{\text{lost}}$  is greater than  $P_j$ 
    retransmit  $P_i^{\text{lost}}$  and drop  $P_j$ 
  else
    send the regular packets and ignore the retransmission
    request for  $P_i^{\text{lost}}$ 
}

```

where  $P_i^{\text{lost}}$  stands for the  $i$ -th lost packet;  $T_{\text{cur}}$  represents the current time;  $D_s$  is a slack term;  $T_d(P_i^{\text{lost}})$  is the deadline for  $P_i^{\text{lost}}$  scheduled to be displayed and  $\text{size}(P_i)$  is the packet size of  $P_i$ ;  $\overline{RTT}_i$  denotes the round-trip delay estimated for the  $i$ -th lost packet  $P_i^{\text{lost}}$ .

To estimate  $\overline{RTT}_i$ , we use the timing information carried in RTP packets and assume that the clock times of server and client are synchronized. The server appends a timestamp to the header of each RTP packet so that the receiver can calculate the time difference according to the timestamp at the receiver by (2) and (3),

$$RTT_k = 2 \times (t_k^r - t_k^s) \quad (2)$$

$$\overline{RTT}_i = \frac{1}{C_i} \sum_{k=1}^{C_i} RTT_k \quad (3)$$

where  $RTT_k$  denotes the round-trip time of packet  $k$ ;  $t_k^r$  and  $t_k^s$  stand for the receiving time at the client and the sending time at

the server, respectively;  $\overline{RTT}_i$  denotes the average  $RTT$  in the interval  $C_i$  which consists a constant number of packets.

The proposed method needs to generate additional side information for recording the loss-impact value of each video packet. In our experiments, the impact values range from 0 to 800, thereby consuming two overhead bytes for each video packet, leading to an average cost of about 0.2% of the packet size. This side information is transmitted to the home server as hints for making retransmission decisions while streaming.

**4. EXPERIMENTAL RESULTS**

In this work, we used a two-state Markov model, which adopts a simplified Gilbert channel at the packet level [6], to generate the channel packet loss patterns as shown in Fig. 4. This model can capture the bursty nature of packet losses in WLAN. In Fig. 2,  $S_0$  indicates a good state, and  $S_1$  is a bad state.  $P_{00}$ ,  $P_{01}$ ,  $P_{10}$ , and  $P_{11}$  represent the state transition probabilities, respectively. In our experiments the packet loss rates (PLR) were set to be 5% and 10%, respectively, and the mean burst length was assumed to be 10 packets.

Two CIF (352x288) test sequences, “Foreman” and “Coastguard,” are encoded with a fixed quantization step-size and a frame rate of 30 Hz with a GOP size of 30 frames (with IPPP... structure) and 4 packets per frame using a public-domain MPEG-4 codec.

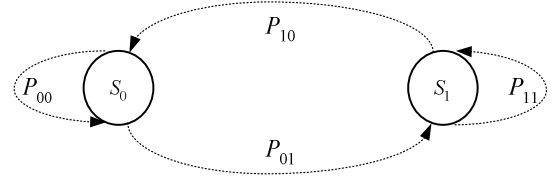


Fig. 4. Two-state Markov channel model.

Table 1 and Figs. 5 and 6 show PSNR performance comparisons for the two test sequences. We compare the proposed scheme with the typical Early-Deadline-First (EDF) scheme [1] and that without retransmission protection. Since the proposed scheme choose to retransmit lost packets with higher impact values by dropping regular packets of less importance, it usually can achieve better trade-offs in terms of video quality. Because the EDF scheme only consider the play-out deadline without taking into account the priority of video packets, the more important packet will probably be dropped should a retransmitted video packet's play-out deadline is earlier than the drop one's. Besides, with the proposed method, if the client does not correctly receive a retransmitted video packet, the dropped packet may still have the chance to be retransmitted again if the delay constraints still can be met. The experimental results show that, in most cases, the proposed scheme achieves significantly better PSNR improvement by about 0.5-1.5 dB than the EDF method, especially for packet losses that occur in high-motion frames. In some cases, the EDF scheme performs even worse than that without retransmission protection.

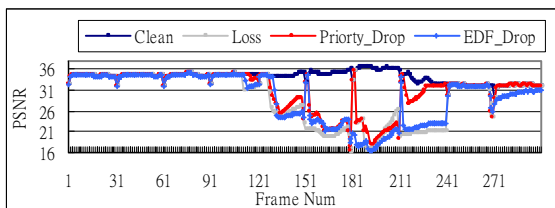
Table 1. Average PSNR comparison of different schemes for two test sequences (mean burst length = 10)

(a) Foreman

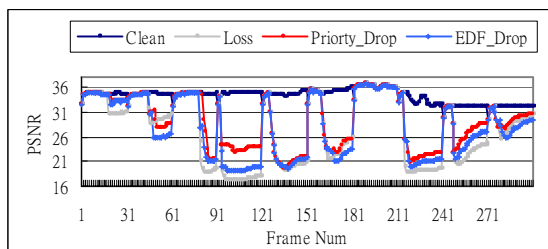
PLR	Clean	No ARQ	Priority_Drop	EDF_Drop
5%	34.11 dB	29.14 dB	30.71 dB	29.28 dB
10%	34.11 dB	27.34 dB	29.00 dB	27.96 dB

(b) Coastguard

PLR	Clean	No ARQ	Priority_Drop	EDF_Drop
5%	32.65 dB	29.09 dB	30.77 dB	28.76 dB
10%	32.65 dB	26.68 dB	27.91 dB	27.39 dB

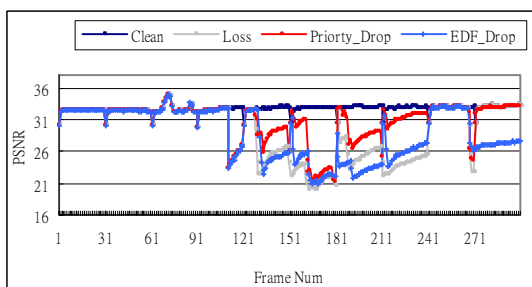


(a) PLR = 5%

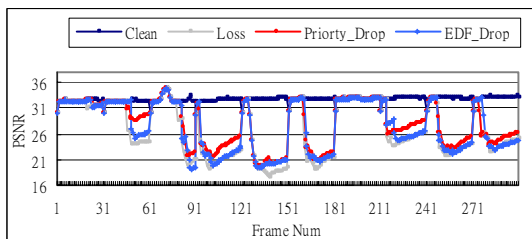


(b) PLR = 10%

Fig. 5. Frame-by-frame PSNR Performance comparison for two packet loss rates (Foreman).



(a) PLR = 5%



(b) PLR = 10%

Fig. 6. Frame-by-frame PSNR Performance comparison for two packet loss rates (Coastguard).

## 6. CONCLUSION

In this work, we have proposed a prioritized retransmission scheme based on the scenario that no dedicated bandwidth is reserved for retransmitting lost packets. The proposed method first estimates the error propagation effect of each lost packet in an off-line coding process. Such information is subsequently used in the home server as a cost for the Greedy algorithm to determine the retransmission policy. Experimental results show that, in most cases, the proposed scheme achieves significantly better average PSNR improvement by about 0.5-1.5 dB than the EDF method, especially for high-motion sequences.

## REFERENCES

- [1] T. Hasegawa, T. Kato, and K. Suzuki, "A video retrieval protocol with video data prefetch and packet retransmission considering play-out deadline," in *Proc. IEEE Int. Conf. Network Protocols*, pp. 32-39, Oct.-Nov. 1996.
- [2] S. H. Kang and A. Zakhor, "Packet scheduling algorithm for wireless video streaming," in *Proc. Workshop Packet Video*, Apr. 2002, Pittsburgh.
- [3] W. Tan and A. Zakhor, "Packet classification schemes for streaming MPEG video over delay and loss differentiated networks," in *Proc. Packet Video Workshop*, Apr. 2001, Kyongju, Korea.
- [4] A. Majumdar, D. Sachs, I. Kozintsev, K. Ramchandran, and M. Yeung, "Multicast and unicast real-time video streaming over wireless LANs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp.524-534, Jun. 2002.
- [5] D. Wu, Y.T. Hou, W. Zhu, Y.Q. Zhang, J.M. Peha, "Streaming video over the Internet: approaches and directions," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, No.3, pp. 282-300,, Mar.2001.
- [6] M. Zorzi, R. R. Rao, and L. B. Milstein, "ARQ error control for fading mobile radio channels," *IEEE Trans. Veh. Technol.*, Vol. 46, pp. 445-455, May 1997.
- [7] W. M. Lam, A. R. Reibman, and B. Liu, "Recovery of lost or erroneously received motion vectors," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Apr. 27-30, 1993, pp. V417-V420, Minneapolis, MN.
- [8] T. H. Cormon, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd Ed., MIT Press, 2001.
- [9] D. Hertrich, *MPEG4 video transmission in wireless LANs: Basic QoS support on the data link layer of 802.11b*, Minor Thesis, Technical University Berlin, 2002, [<http://www.daniel-hertrich.de/studienarbeit/>].
- [10] *Information technology- Telecommunications and information exchange between systems- Local and metropolitan area networks—specific requirements- Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications*, IEEE 802.11 stand, 1997.