

2014 AASRI Conference on Sports Engineering and Computer Science (SECS 2014)

## Retransmission and Flow Control for a Video Real-Time Transport Protocol

Tzu-Hung Lin<sup>a</sup>, Cheng-Yu Yeh<sup>b</sup>, Shaw-Hwa Hwang<sup>c\*</sup>

<sup>a</sup>NTUT, Electrical Engineering, #1, Sec. 3, Chung-hsiao East Rd., Taipei, Taiwan, ROC

<sup>b</sup>NCUT, Electrical Engineering, #57, Sec. 2, Zhongshan Rd., Taichung, Taiwan, ROC

<sup>c</sup>NTUT, Electrical Engineering, #1, Sec. 3, Chung-hsiao East Rd., Taipei, Taiwan, ROC

---

### Abstract

This study provides a method to mitigate the effect of packet loss and congestion for a real-time transport protocol (RTP) [1] for video streaming. The sequence number of the RTP [2] was used to detect packet loss such that retransmission [3] and flow control [4] [5] could be addressed. Retransmission ensures that an entire frame can be appropriately decoded. Users do not see the broken image, and do not waste bandwidth. Flow control addresses the rate of data transmission that is adjusted dependent on the available network bandwidth. Frames per second are reduced by low bandwidth, but an image can still be viewed appropriately, without a black screen or freeze-frame occurring.

© 2014 The Authors. Published by Elsevier B. V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/3.0/>).

Peer-review under responsibility of Scientific Committee of American Applied Science Research Institute

**Keywords:** congestion, flow control, packet loss, real-time transport protocol (RTP), retransmission, video streaming.

---

### 1. Introduction

Numerous multimedia network protocols have been developed. In the communication part, such as SIP, RTSP, H.323, and HTTP, and in the multimedia data transmission part, such as the real-time transport protocol (RTP) and HTTP. SIP, RTSP, and H.323 all use RTP to transmit multimedia data, and primarily use

---

\* Corresponding author. Tel.: +886972015239;

E-mail address: [ceepoo711@hotmail.com](mailto:ceepoo711@hotmail.com).

a user datagram protocol (UDP). UDPs exhibit packet loss and congestion problems, but the RTP does not include standards of retransmission and flow control. Moreover, most RTP applications do not implement a retransmission mechanism; therefore the effect on voice streaming is slight, but that on video streaming is severe.

Because bandwidth is insufficient, only voice data have been transferred; however, the effect of packet loss on voice is small and communication is possible when packet loss is not excessive. Furthermore, because voice data are small, a flow control mechanism is unnecessary.

Increasing bandwidth has allowed video streaming to become mainstream. Video data are large and, therefore, require division into multiple packets; however, frames cannot be decoded appropriately when one packet has been lost. This prevents users from seeing an image appropriately and also wastes bandwidth. Flow control mechanisms are therefore critical.

## 2. Flow control

Once a sender starts sending video streaming data, a receiver sends a PKL (Packet Loss) packet to the sender to confirm the current network state every 100 millisecond. This allows the sender to control the rate of data transmission.

Before the sender sends a frame, the sender calculates how long a frame takes to send at the available bandwidth.

$$\text{SendTime(sec)} = \frac{\text{DataLen(bit)}}{\text{BitRate(bit/sec)}}$$

### 2.1. Bit Rate Reduction

The bit rate changes in accordance with the current network state. The sender reduces the bit rate by half when packet loss exceeds 5% per second.

$$\text{BitRate} = \text{BitRate} \times \left(1 - \frac{\text{PKL}}{\text{ByteRate} \times 2}\right)$$

$$\text{Byte Rate} = \frac{\text{Bit Rate}}{8}$$

The size of one packet is 1 kb (1 PKL = 1 kb).

When transmitting at a new bit rate, the bit rate is reduced by half if the PKL exceeds 240.

### 2.2. Bit Rate Increase

The sender increases the bit rate when packet loss is 0% per second.

$$\text{BitRate} = \text{BitRate} + \delta e^{\alpha}$$

$$\delta = \frac{\text{MaxBitRate}}{100}$$

$$1 \leq \alpha < \text{Threshold}$$

$$\text{Threshold} = \frac{\text{Current Bit Rate} + \text{Bit Rate just before loss event}}{2}$$

### 3. Retransmission

Once a sender initiates the sending of video streaming data, the receiver determines the packet loss based on the sequence number in the RTP header. When packet loss occurs, the receiver sends the RTP packet to the sender. The RTP packet contains the same sequence number as the lost packet, but does not contain payload data. The sender can determine which RTP packet is lost and retransmit the RTP packet in the next time slice, when receiving the empty RTP packet. The detailed retransmission mechanism is presented in Fig. 1.

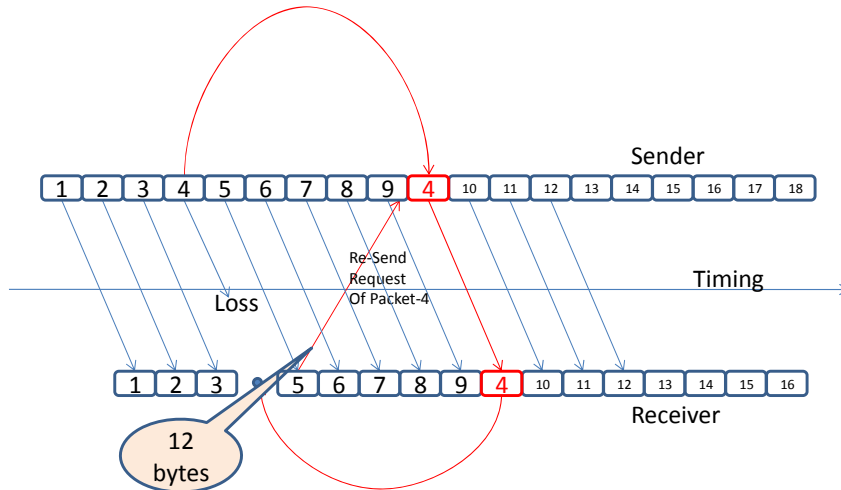


Fig.1. Schematic diagram of retransmission.

Using the above method still incurs a problem; after packet loss occurs, successful retransmission must be confirmed before subsequent frames are sent. However, waiting for a packet halts the function of the system. Thus, the receiver sets a timer for each retransmission request and discards the frame when the timer runs out. The system can continue normally without decoding the unnecessary frame, preventing a broken image from appearing and a performance overhead from being incurred.

#### 3.1. Comparisons with RFC4588 (RTP Retransmission Payload Format)

RFC4588 provides an RTP payload format for performing retransmission. Compared with the proposed method, RFC4588 exhibits several weaknesses. First, it requires an RTCP (Real Time Control Protocol), which increases the complexity of the system and consumes performance. Furthermore, it requires another channel to perform RTP retransmission, such that a RTP streaming requires three channels. Finally, it uses SDP to perform retransmission requests, the format of which is complex and exhibits a large packet size, increasing the parsing time and bandwidth requirements. The differences between RFC4588 and the proposed method are listed in Table 1.

Table 1. The different with RFC4588 and we proposed method

	RFC-4588 RTP Retransmission Payload Format	We proposed method
Related protocol	Need to use with RTCP	Do not need other protocol
Channel number for each RTP streaming	Three channel (RTP, Retransmission RTP, RTCP)	One channel
Retransmission requests	Use SDP (Tens to hundreds bytes)	Use RTP (12 bytes)

#### 4. Experimental results

This section presents the effect of the retransmission and flow control mechanism. IP camera was used to transmit video data over the Internet and we count frame numbers that can be decoded appropriately.

- Codec type: MPEG4
- Video Size = 640\*480 (VGA)
- Frame Rate = 30 fps
- Bandwidth = 2 Mbps

First, the effect of retransmission was tested. The retransmission mechanism increased the received complete FPS by 2.66 times (from 10.8 to 28.8) compared with no retransmission. The detailed experimental results are listed in Table 2.

Table 2. Test results of Retransmission

Retransmission	Received complete Frame/Second
X	10.827
V	28.8

Next, the flow control mechanism was implemented and NAT (Network Address Translation) was used to limit the bandwidth and test the effect of flow control at various bandwidths. The received complete FPS exerted a positive effect at each bandwidth, as shown in Tables 3 and 4.

Table 3. Test results of Flow-Control

Flow-Control	Retransmission	Bandwidth		
		128K	256K	512K
X	V	0.463	1.465	3.674
V	V	1.595	2.936	5.753

Table 4. Test results of Flow-Control continue

Flow-Control	Retransmission	Bandwidth	
		1M	2M
X	V	8.017	25.893
V	V	10.305	27.708

## 5. Conclusion

This paper presents a retransmission and flow control method. Retransmission can reduce the effect of packet loss on video streaming by using a small number of retransmission packets. Flow control controls the sending rate based on the number of packets lost, so that the sending rate adheres to the available bandwidth. Retransmission and flow-control improves the performance of video streaming, and users can clearly appreciate the difference when viewing images.

## References

- [1] R Frederick, and V Jacobson 2003. Rtp: A transport protocol for real-time applications.
- [2] C. Bormann, C. Burmeister, M. Degermark, H. Fukushima, H. Hannu, L-E. Jonsson, R. Hakenberg, T. Koren, K. Le, Z. Liu, A. Martensson, A. Miyazaki, K. Svanbro, T. Wiebke, T. Yoshimura, and H. Zheng 2001. RObust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed.
- [3] Nick Feamster and Hari Balakrishnan. 2002. Packet Loss Recovery for Streaming Video.
- [4] Bolot, J.-C., INRIA, Sophia-Antipolis, France, and Turetti, T. 1994. A rate control mechanism for packet video in the Internet.
- [5] Shanwei Cen, Jonathan Walpole, Calton Pu. 1998. Flow and congestion control for Internet media streaming applications.