# ON COMPENSATION TECHNIQUE IN MULTIMEDIA STREAMING SYSTEM

*Yongfeng Li and Kenneth Ong*

Department of Electrical and Computer Engineering
National University of Singapore, 4 Engineering Drive 3, Singapore 117576
Email: eleongk@nus.edu.sg

**ABSTRACT**

We describe a compensation method for real-time multimedia streaming to improve the user-perceived quality of video. Based on TCP Friendly Rate Control (TFRC), this method takes advantage of the resources available at the proxy server and considers the content-aware principle as well. Using feedback from the client, the proxy server dynamically drops or compensates the enhancement layers of the MPEG-4 stream. We optimize performance by carefully choosing the compensation point to maximize the compensation effect, while at the same time meeting the constraints of the client buffer and real-time requirements. Compared to the traditional prefetch method, the proposed method can further improve the video quality which is demonstrated through extensive simulations.

## 1. INTRODUCTION

Technology advances in real-time multimedia transport over networks have given rise to many multimedia applications which are closely related to the development of underlying protocols and the efficient usage of network and server resources. In order to support more clients, proxy servers are widely used in streaming systems [1] [2]. They can cache recently requested streams or a prefix of the streams to reduce server load and response time.

Since the Internet does not provide QoS guarantee mechanisms, employing proper protocols for the multimedia stream is important. Different from traditional applications, TCP is not suitable for real-time multimedia streaming because of its retransmission mechanisms. And UDP is also not appropriate because of its greedy property. Fortunately, TFRC proposed in [3] [4] can fit the streaming system very well. And the client side information provided by TFRC can enable us to do scalable rate control more efficiently. We believe that the more information we get, the more intelligently the streaming system will be [5].

In practice, the transmission rate of the multimedia stream is constrained by the available bandwidth and the maximum decoding ability of the client. When the bandwidth is not sufficient to transmit full size or full frequency streams that the client requests, filter mechanisms [6] [7] should be employed to drop the MPEG-4 enhancement layer data. On the other hand, when bandwidth is more than sufficient, extra bandwidth can be utilized to compensate previously dropped data. However, the real-time property requires those compensated data to arrive at the client before their corresponding base layer is displayed. And the client buffer should never overflow. Thus we optimize the performance of our proposed technique by choosing the optimal compensation point to make the compensation effect maximum while meeting these constraints. Since real-time transport of video is the predominant part of multimedia, we are only concerned with video streaming. However, the method can be applied to scalable multimedia streaming as well.

The paper is organized as follows. Section 2 gives some background information about the TFRC protocol. Section 3 provides a general problem overview and presents its mathematical model. Section 4 derives the optimal solutions and they are evaluated via simulation in Section 5. Section 6 concludes the paper in the end.

## 2. RATE CONTROL WITH TFRC

TFRC is a congestion-control mechanism that enables a non-TCP session to behave in a TCP-friendly manner. The TFRC sender estimates the throughput of a TCP session sharing the same path using following equation:

$$X = \frac{s}{R\sqrt{\frac{2bp}{3}} + t_{RTO}(3\sqrt{\frac{3bp}{8}})p(1+32p^2)}$$

$X$ is the transmit rate in bytes/second. $s$ is the packet size in bytes. $R$ is the round-trip time in seconds. $p$ is the loss event rate, between 0 and 1.0, of the number of loss events as a fraction of the number of packets transmitted. $t_{RTO}$ is the TCP retransmission timeout value in seconds. $b$ is the number of packets acknowledged by a single TCP acknowledgement. In our experiment, the information is obtained by exchanging RTCP messages between the RTCP Sender of proxy server and client application. In addition, the fullness information of the client buffer ($b(t)$) is attached to the RTCP messages.

*0-7803-9332-5/05/$20.00 ©2005 IEEE*

## 3. GENERAL PROBLEM DESCRIPTION

A proxy-assisted video delivery system shown in [1] is used in this paper. A central video server delivers video streams to a large number of clients through proxy servers. Those proxy servers respond to clients' requests quickly and at the same time receive data from the central server. The bandwidth between proxy server and clients may drop greatly during transmission and the enhance-ment layer data may be dropped. When the bandwidth resumes and exceeds the decoding ability of clients, the compensation method can be used to increase the user-perceived quality. Since the more data is transmitted, the higher the user-perceived quality is [7], our objective is to maximize the compensation effect (*Ce*) defined by

$$Ce = \frac{Compensated\_Data}{Dropped\_Data + Compensated\_Data}$$

Then the optimal compensation point is derived to maximize *Ce* while at the same time meeting real-time and client buffer constraints.

Table 1: Parameters of compensation model

| | |
|---|---|
| $D$ | Maximum decoding rate of client |
| $C$ | Client buffer size |
| $d(t)$ | Consuming rate of client buffer at time $t$ |
| $B(t)$ | Available bandwidth at time $t$ |
| $x(t_0,t_1)$ | Dropped bytes between time $t_0$ and $t_1$ |
| $b(t)$ | Content volume of client buffer at time $t$ |
| $S(t)$ | Transmission rate at time $t$ |

Table 1 summarizes the parameters of the following mathematical model. We assume that they are a-priori information or provided by the underlying protocols. For ease of exposition, we focus on the interaction between one proxy server and one client. Our results, however, are applicable to the multi-client case.

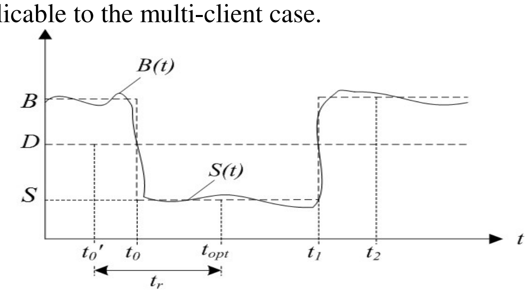


Figure 1: Relation between transmission rate and bandwidth

For simplicity and safety, we consider the maximum decoding rate of the client as constant $D$ and the bit rate of the video object with full size and full frequency on the proxy server is higher than $D$. Thus in Fig. 1, the proxy server drops part of the enhancement layer data before time $t_0$. At time $t_0$, the proxy server senses a drop in the available bandwidth. Then it drops more data to achieve a lower transmission rate, for example $S$. Later at time $t_1$, the proxy server senses a rise in the bandwidth. The extra bandwidth above $D$ can be utilized to compensate for the possible video quality decrease between time $t_0$ and $t_1$. However, it may not be desirable to retransmit all the dropped data ( $x(t_0,t_1) = \int_{t_0}^{t_1} (D - S(t))dt$ ) because when the enhancement layer of time $t_0$ reaches the client, the base layer of time $t_0$ may have already been decoded and displayed. So we need to choose a point $t_{opt}$ from which we begin to transmit the dropped data and ensure that these data arrive at the client in time. Thus the time to transmit dropped data between time $t_{opt}$ and $t_1$ should be less than the time for the client to display the residual data in its buffer before time $t_{opt}$. The compensation process is terminated at time $t_2$ after which the transmission rate is back to $D$. Thus, we get:

$$\frac{\int_{t_{opt}}^{t_1} (D - S(t))dt}{B(t) - D} \le \frac{b(t_1) - \int_{t_{opt}}^{t_1} S(t)dt}{d(t)} \quad (1)$$

Besides, in order to prevent client buffer overflow, we get:

$$b(t_1) + \int_{t_1}^{t} (B(t) - d(t))dt \le C \text{ (for every } t_2 \ge t \ge t_1) \quad (2)$$

So the problem can be summarized as finding the mini-mum $t_0 < t_{opt} < t_1$ subject to the constraints (1) and (2).

## 4. STREAMING DATA COMPENSATION

To solve the optimization problem and considering the practical issues, we assume that the available bandwidth does not fluctuate greatly most of the time (except at time $t_0$ and $t_1$). So $B(t)$ is written as constant $B$ after time $t_1$, and $S(t)$ is expressed as constant $S$ between time $t_0$ and $t_1$. It is also noted that the consumption rate of the client buffer changes according to the different values of $b(t)$ because of the constant frame display rate. For example, the consumption rate of the data that have been received during the period before $t_0$ is $D$, while between $t_0$ and $t_1$, the rate is $S$. Thus according to $b(t)$, we classified $d(t)$ into the following two situations:

A. If $b(t_1) \le (t_1 - t_0) \cdot S$ , $d(t) = S$

B. If $b(t_1) > (t_1 - t_0) \cdot S$ , $d(t) = S$ when decoding those data client received between $t_0$ and $t_1$; $d(t)=D$ otherwise.

### 4.1 Client Buffer is Not Full

We consider situation A first when the client buffer is not full and its consumption rate is $S$ at the time of compen-sation. Thus the constraint (1) & (2) can be rewritten as:

$$(t_1 - t_{opt}) \cdot (D - S)/(B - D) \le (b(t_1) - (t_1 - t_{opt}) \cdot S)/S \quad (3)$$

$$b(t_1) + (t - t_1) \cdot (B - S) \le C \text{ (for every } t_2 > t > t_1) \quad (4)$$

We choose the time that the client buffer is the fullest and prevent its overflow at this time so that constraint (4) is satisfied all the time. The time $t_2$ when the compensation process is finished is such a time point. It is obvious from Fig. 1 that before time $t_2$, the input rate of the client buffer is larger than its consumption rate. After time $t_2$, these two rates become the same value $D$. Thus, we get the following equation:

$$(B-D)\cdot(t_2-t_1)=(t_1-t_{opt})\cdot(D-S)$$

So the value of $t_2$ is derived as:

$$t_2=t_1+(t_1-t_{opt})\cdot(D-S)/(B-D) \quad (5)$$

Then, after substituting $t$ in constraint (4) with $t_2$ using equation (5), we get

$$t_{opt}\ge t_1-\frac{(C-b(t_1))\cdot(B-D)}{(D-S)\cdot(B-S)} \quad (6)$$

The constraint (3) can also be simplified as

$$t_{opt}\ge t_1-\frac{b(t_1)\cdot(B-D)}{(B-S)\cdot S} \quad (7)$$

So from the above constraints (6) and (7), the optimal compensation time is

$$t_{opt}=\max[t_1-\frac{(C-b(t_1))\cdot(B-D)}{(D-S)\cdot(B-S)},t_1-\frac{b(t_1)\cdot(B-D)}{(B-S)\cdot S}]$$

### 4.2 Client Buffer is Full

Now we consider situation B when the client buffer is almost full at compensation time. At time $t_1$, the client consumes the data received at time $t_0'$. Since $b(t_1)>(t_1-t_0)S$, $t_0'$ must be less than $t_0$. In Fig. 1, we observe that $d(t)$ is equal to $D$ when the client is consuming the data received between $t_0'$ and $t_0$. And $d(t)$ is equal to $S$ when consuming the data the client received between $t_0$ and $t_{opt}$. So the constraint (1) can be rewritten as

$$(t_1-t_{opt})\cdot(D-S)/(B-D)\le t_r \quad (8)$$

where $t_r$ is the time that the client displays the residual data in its buffer before compensation point $t_{opt}$. The data that is displayed in this period is $b(t_1)-(t_1-t_{opt})\cdot S$ . It is noted that those data are consumed at a different rate. There $b(t_1)-(t_1-t_{opt})\cdot S-(t_{opt}-t_0)\cdot S$ is consumed at rate $D$ and the remaining $(t_{opt}-t_0)\cdot S$ data is consumed at rate $S$. So $t_r=[b(t_1)-(t_1-t_{opt})S-(t_{opt}-t_0)S]/D+(t_{opt}-t_0)$.

Besides, depending on the volume of data that the client has consumed in the compensation process, constraint (2) is considered in two parts as:

$$b(t_1)+(t-t_1)\cdot B-(t_0-t_0')\cdot D-(t_{opt}-t_0)\cdot S\le C \quad (9)$$

$$b(t_1)+(t_1-t_{opt})\cdot(D-S)\le C \quad (10)$$

Since $t_0'=(t_{opt}-t_r)$, we substitute it into constraint (9) and replace $t$ with $t_2$ using expression (5) due to the same reason. Thus, by solving constraints (8) (9) and (10), we get the optimal (minimum) compensation point

$$t_{opt}=\max[t_x,t_1-\frac{C\cdot(B-D)}{D\cdot(B-S)},t_1-\frac{C-b(t_1)}{D-S}]$$ , where $t_x$ = $[(D^2+SB-2SD)t_1+(DB-D^2-SB+SD)t_0-(B-D)b(t_1)]/[(B-S)D]$

## 5. PERFORMANCE EVALUATION

### 5.1. Simulation Model

To examine the performance of the proposed method, we developed the MPEG-4 proxy server and client in NS2. The proxy server reads and sends the MPEG-4 video stream to the client based on the TFRC protocol with the compensation method. The network architecture shown in Fig. 2 is used for simulation. All the queues use the FIFO drop-tail scheduling discipline. The link delay is set to 20*ms* between two switches (node 4 and 5) and 10*ms* between a switch and an end system. The TCP connections are modeled as FTP flow that always has data to send and last for the entire simulation time. Its transmission rate is 300kb/s. The UDP connections are modeled as CBR flow to simulate the available bandwidth drop for MPEG-4 stream session. Its transmission rate is 700kb/s. The client buffer size is 3.5MBytes. And the simulation runs for 60 seconds which is long enough to evaluate the performance.

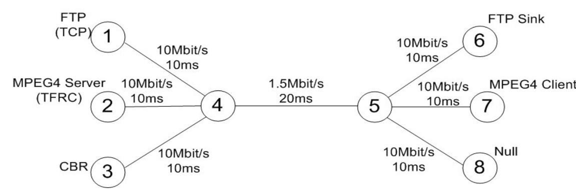

Figure 2: Simulation topology

### 5.2. Simulation Scenarios

We perform simulations in the following scenarios:

**Scenario 1**: We simulate the whole transmission process under two situations explained in Section 4. The optimal compensation points ($t_{opt}$) are calculated and the simulation results are shown in Fig. 3.

**Scenario 2**: From Fig. 3, it is obvious that the fullness degree of the client buffer at the time of compensation influences *Ce*. So we get their relation in Fig. 4.

**Scenario 3**: Similarly, the influence of the client buffer size on *Ce* is also simulated. The simulation results in two situations (Section 4.1 and 4.2) are shown in Fig. 5.

**Scenario 4**: MPEG-4 video stream is simulated to be transmitted using the compensation method, prefetch and normal transmission. The decoded PSNR on the client are compared. And the MPEG-4 traffic is obtained from the MPEG-4 trace file presented in [8]. The simulation parameters and simulation results are shown in Fig. 6.

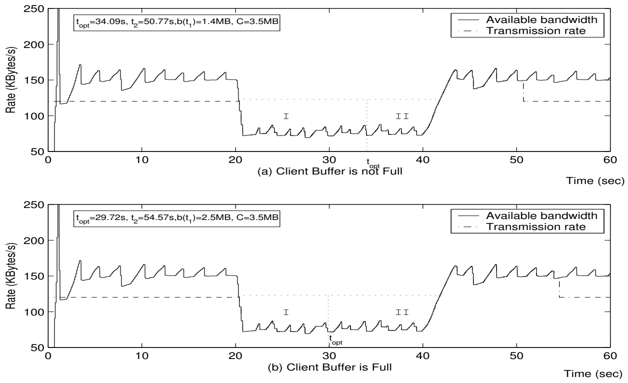


Figure 3: Available bandwidth and transmission rate curves during compensation transmission

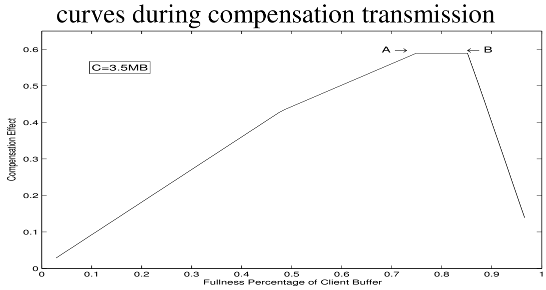


Figure 4: Impact of fullness degree of client buffer on *Ce*

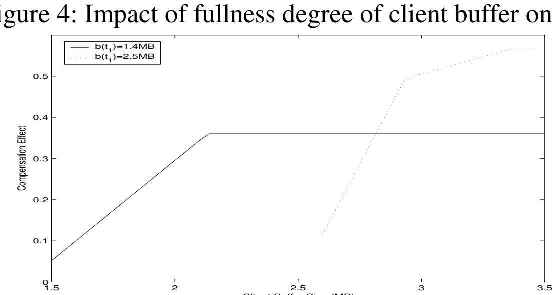


Figure 5: Influence of client buffer size on *Ce*

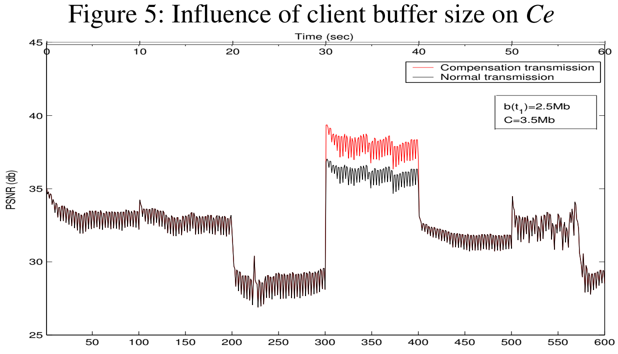


Figure 6: Comparison of PSNR between normal transmission and compensation transmission

### 5.3. Results Analysis

Fig. 3 shows the available bandwidth and transmission rate during simulation. Regions I and II are the volume of dropped data if the stream is transmitted without the compensation method. Region II alone is the volume of data that can be compensated with our method.

The second measurement concerns with the influence of degree of fullness of the client buffer on *Ce*. From Fig. 4, we can see that *Ce* is not linear with the free client buffer space. Before point A, *Ce* will increase because the more data inside the client buffer, the more time is needed to compensate dropped data. However, after the point B, *Ce* will decrease because the client buffer does not have enough free space to accommodate the extra compensation data.

Similarly, Fig. 5 shows that blindly increasing the client buffer size may not achieve better *Ce* if residual data inside the client buffer at compensation time is fixed.

The last measurement in scenario 5 further justifies the effectiveness of the compensation method. Fig. 6 shows that it can increase the video quality by 2.5db for about 100 frames. Since the prefetch technique can only prevent later drop of data due to bandwidth decrease, it has no effect on previous dropped data in this scenario. So its curve overlaps the curve of normal transmission.

## 6. CONCLUSIONS

We described a new method to deal with unpredictable bandwidth decrease during multimedia streaming over network. The video quality is improved according to the client and network resource availability. Combined with the TFRC protocol and the content-aware principle, the proposed delivery method can compensate streaming data significantly. Simulation results demonstrate the improvement with respect to user-perceived video quality.

## 7. REFERENCES


[1] L. Gao, Z. Zhang, D. Towsley, “Proxy-Assisted Techniques for Delivering Continuous Multimedia Streams,” *IEEE/ACM Trans. Networking*, vol. 11, pp. 884–894, Dec. 2003.

[2] H. Fahmi, M. Latif, S. Sedigh–Ali, A. Ghafoor, P. Liu, L. H. Hsu, “Proxy Servers for Scalable Interactive Video Support,” *IEEE Comput.*, vol. 34, pp. 54–60, Sept. 2001.

[3] J. Padhye, J. Kurose, D. Towsley, R. Koodli, “A model based TCP-friendly rate control protocol,” in *Proc. NOSSDAV 99*, Jun. 1999.

[4] M. Handley, S. Floyed, J. Padhye, J. Widmer, “TCP Friendly Rate Control (TFRC): Protocol specification,” Internet Request for Comments 3448, Jan. 2003.

[5] S. F. Chang, P. Bocheck, “Principles and Applications of Content-aware Video Communications,” in *Proc. IEEE Int. Symp. Circuits and Syst.*, May 2000.

[6] N. Yeadon, F. Garcia, D. Hutchison, D. Shepherd, “Filters: QOS Support Mechanisms for Multipeer Communications,” *IEEE J. Select. Areas Commun.*, vol. 14, No. 7, pp. 1245–1262, Sept. 1996.

[7] Y.F. Li, K. Ong, “Optimal Proxy Filter for Multimedia Streaming over Network,” Technical Report, National University of Singapore, 2004.

[8] P. de Cuetos, M. Reisslein, K.W. Ross, “Evaluating the Streaming of FGS–Encoded Video with Rate-Distortion Traces” (http://www.eurecom.fr/decuetos), June 2003