TSFD: Two Stage Frame Dropping for Scalable Video Transmission over Data Networks *

Bing Zheng
Dept. of Electrical and Computer Eng.
University of Dayton
Dayton, OH 45469
zhengbin@flyernet.udayton.edu

Mohammed Atiquzzaman
School of Computer Science
University of Oklahoma
Norman, OK 73019
atiq@ieee.org

Abstract

Scalable video transmission is used to adjust the rate of video depending on the level of network congestion. Previous studies on scalable video transmission of MPEG over ATM ABR service required major changes in the network protocols, and did not provide methods to determine the ABR connection parameters. In this paper, we propose a new scalable video transmission scheme which does not require major changes in network protocols. In our proposed scheme, frames are dynamically dropped either by the source or the network depending on the level of network congestion.

1 Introduction

Video transmission requires a large bandwidth from the network. During periods of network congestion, the bit rate of video needs to be scaled dynamically depending on the level of network congestion [1, 2]. This paper deals with a new video scaling scheme for interactive video on demand systems.

The authors in [2] investigated layer based scalable MPEG video transmission scheme over ATM networks. However, the schemes have to be implemented at the *video source coder*. The authors in [1, 3] studied *static layer based and slice based scalable* MPEG transmission over the ATM VBR/ABR hybrid service. Unfortunately, their schemes *need modification of the the standard AAL5 protocol*, and hence is not suitable for practical deployment. Moreover, they *did not provide any algorithm to choose an optimal value of Minimum Cell Rate (MCR)* to ensure QoS. The *objective of this paper* is to develop a scalable scheme for transmitting stored MPEG video over a bandwidth limited channel without requiring major changes of the standard network protocols.

We propose a novel Two Stage Frame Dropping (TSFD) Scheme for scalable MPEG transmission over an ATM ABR service. In addition to frames being dropped by the server in the case of network congestion, the server also marks low priority frames to be dropped by the network in the case of severe congestion. An important *contribution* of this paper is that the value of MCR *is chosen by a combination of the bit rate and burstiness of video*. Our proposed TFSD scheme is based on encapsulating video frames with priority information which is used to drop frames by the network during congestion. The scheme requires *no major change of network protocols*.

The effect of MPEG video GoP on the client buffer size has been also analyzed. A general framework has have been developed to determine the client buffer size for no overflow at the client.

The rest of this paper is organized as follows. The principle of TSFD is presented in Section 2, followed by Section 3 where we develop a framework to determine the optimal buffer size. Numerical results are presented in Section 4, followed by conclusions in Section 5.

2 Two Stage Frame Dropping (TSFD)

The *Two Stage Frame Dropping* (TSFD) scheme proposed in this paper consists of two parts. The first is the dynamic priority encapsulation and frame discarding procedure under different network congestion. The second is the adaptive choice of MCR by taking into consideration the burstiness and bit rate of video. Notations used throughout the paper are given below.

f: MPEG video frame rate in frames/second;

n: Distance between I frames for MmNn GoP;

m: Distance between P frames for MmNn GoP;

X_I, X_P, X_B: Average size of I, P and B frames in bits;

x_I: I frame size in bits;

β_I: Average bit rate of I frames defined as $X_I f$;

β_P: Average bit rate of P frames defined as $X_P f$;

β_B: Average bit rate of B frames defined as $X_B f$;

$E_0[\beta]$: Average bit rate of video stream with I, P and B frames;

$E_1[\beta]$: Average bit rate for video stream with I and P frames;

$E_2[\beta]$: Average bit rate for video stream with only I frames;

*This work was supported by a fellowship from DAGSI

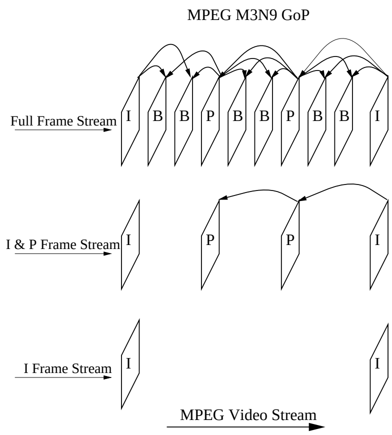

Figure 1: Three video streams having different combinations of frame types.

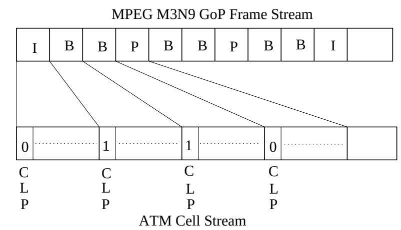

Figure 2: Priority encapsulation in the TSFD scheme.

k: Speed factor for fastforward / fastbackward (FFW/FBW) operation in a video on demand system;

ACR, MCR: The available and minimum cell rates respectively during playback;

C_c: Critical value of client buffer size;

ρ: The ratio of MCR to $E_2[\beta]$;

T_d: Fixed Round Trip Time (FRTT) from server to client;

τ_f: Duration of FFW/FBW operation;

τ_c: Duration of network congestion;

$b = \frac{\beta_I}{\beta_B}$: Burst coefficient for MPEG;

n_1, n_2: The expected number of requests before the server gets the FFW/FBW and playback bandwidth respectively.

2.1 Principle of TSFD

Loss of I, P, or B frames have different effects on video quality. I frame is the reference frame which is most important, P and B frames are less important because they contribute mainly to the improvement of space and time resolution. Therefore, in case of network congestion, B and P frames can be discarded to reduce the bandwidth requirement of video. As shown in Figure 1, different combination of frame types require different amounts of bandwidth. Our proposed TSFD scheme can be described as follows:

- During normal playback, frames are dropped either by the server and/or by the network depending on the level of congestion.
- The assignment of the Cell Loss Priority (CLP) bit (in the ATM cell header) by the server is done as follows. If the bandwidth offered by the network meets the requirements for transporting full frame video stream , the server does not drop frames, and sets CLP=1 for B frames (Figure 2). However, if the bandwidth offered by the network meets the requirements for transporting only I and P frames, the server discards the B frames, and assigns CLP=1 to P frames. If the bandwidth offered by the network just meets the requirements for transporting I frames, the server discards B and P frames; it randomly sets some of the I frames with CLP=1 allowing the network to drop those I frames in the case of congestion.
- In the FFW or FBW mode, the server discards all B and P frames, and sends only I frames to reduce the bandwidth.
- During FFW/FBW operation, the server accepts whatever rate is available from the network, but keeps sending in rate RM cells to request a rate which is no less than the required rate for FFW/FBW operation.

2.2 Adaptive Choice of MCR

Once an ABR connection is set up, any bandwidth request which is higher than the MCR is approved by the network with some probability. We describe below an algorithm to adaptively choose a value of MCR. From Figure 1, the average bit rate corresponding to the three frame streams can be expressed as:

$$E_0[\beta] = \frac{\beta_I + \beta_P(n/m - 1) + \beta_B n/m(m-1)}{n} \tag{1}$$

$$E_1[\beta] = \frac{\beta_I + \beta_P(n/m - 1)}{n} \tag{2}$$

$$E_2[\beta] = \frac{\beta_I}{n} \tag{3}$$

To ensure adequate quality of video at the client, the value of MCR should be between $E_2[\beta]$ and $E_0[\beta]$, and is given by.

$$MCR = \rho E_2[\beta] \tag{4}$$

where ρ has three different values corresponding to the three different streams.

$$\rho_{low} = 1 \tag{5}$$

BEGIN *Connection setup*
Set $MCR = E_0[\beta]/\rho_{low}$
Send ABR Connection Setup Signal
If *Returned MCR* < *MCR*, then Set $MCR = E_0[\beta]/\rho_{middle}$
Send ABR Connection Setup Signal
If *Returned MCR* < *MCR*, then Set $MCR = E_0[\beta]/\rho_{high}$
Send ABR Connection Setup Signal
If *Returned MCR* < *MCR*, then Wait a random time period and try again
END

Figure 3: The algorithm for choosing MCR.

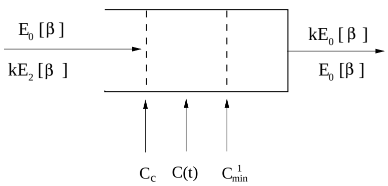

Figure 4: Illustration of C^1_{min} and C_c at client buffer.

$$\rho_{middle} = 1 + \frac{1}{b}\frac{\beta_P}{\beta_B}\left(\frac{n}{m} - 1\right) \tag{6}$$

$$\rho_{high} = 1 + \frac{1}{b}\frac{\beta_P}{\beta_B}\left(\frac{n}{m} - 1\right) + \frac{1}{b}\frac{n}{m}(m-1) \tag{7}$$

ρ reflects the contributions from the B and P frames, and hence is a measure of the burstiness of video. The algorithm to choose MCR is shown in Figure 3.

3 Client Buffer Size

We define the *critical client buffer size* C_c as the size below which the client buffer overflows with a high probability. To determine C_c, we first need to determine the *minimum client buffer fill level* which is defined as the buffer level at which there is no starvation at the client.

3.1 Minimum Client Buffer Fill Level

The value of the MCR that is accepted by the network may be lower than the average rate of the video which gives rise to the following two cases.

- *Case 1:* In FFW/FBW operation, there should be no starvation at the client buffer;
- *Case 2:* In normal playback, there should be no starvation at the client buffer during network congestion.

3.1.1 Case 1: FFW/FBW Operation

As shown in Figure 4, we assume that at time t, the client sends a FFW/FBW request to the video server. The client starts its FFW/FBW operation, consuming video data at a rate $kE_0[\beta]$. We also assume that FFW/FBW lasts for a time duration τ_f. During this period, the client will consume $Q^1_{out} = \int_t^{t+\tau_f} kE_0[\beta]dt$ amount of data. On other hand, because of network congestion and propagation delay, the client will not immediately receive data at the FFW/FBW rate after sending the FFW/FBW request. The delay consists of three parts: $T_d/2$ for the FFW/FBW request to arrive at the server, $n_1 T_d$ to obtain the requested bandwidth of $kE_2[\beta]$, and $T_d/2$ for the data to arrive at the client. During this time, the data input rate to client buffer is still at the rate of $E_0[\beta]$. The input data is denoted by $Q^1_{in1} = \int_t^{t+(n_1+1)T_d} E_0[\beta]dt$. $Q^1_{in2} = \int_{t+(n_1+1)T_d}^{t+\tau_f} kE_2[\beta]dt$ is the amount of input data at FFW/FBW speed. Therefore, for no starvation at the client buffer, the amount of data consumed by the client must be equal to or less than the sum of the arriving data and the previously stored data in the buffer:

$$C(t) + Q^1_{in1} + Q^1_{in2} - Q^1_{out} \geq 0 \tag{8}$$

where $C(t)$ is client buffer fill level at time t. By writing Equation (8) in average value form:

$$C(t) + (n_1+1)T_d E_0[\beta] + (\tau_f - (n_1+1)T_d)kE_2[\beta] - \\ (n_1+1)T_d kE_0[\beta] - (\tau_f - (n_1+1)T_d)kE_0[\beta] \geq 0 \tag{9}$$

Therefore, to prevent the client buffer from starvation during the FFW/FBW operation, the client buffer must have a minimum fill level of C^1_{min}. Note that C^1_{min} is the value of $C(t)$ for the minimum case in Equation (9).

$$C^1_{min} \geq (k-1)(n_1+1)T_d E_0[\beta] + \\ k(\tau_f - (n_1+1)T_d)(1 - \frac{1}{\rho_{high}})E_0[\beta] \tag{10}$$

3.1.2 Case 2: Heavy Network Congestion

In case of heavy network congestion, $ACR = MCR$. Assume that the *congestion duration* lasts for a period of τ_c. To prevent the client buffer from underflow, a minimum fill level of C^2_{min} is required to compensate the difference between the low video data input rate and high data consumption rate from the client buffer as shown in Figure 5.

In the worst case, only after the end of heavy congestion, can the server obtain the required normal playback rate after an average of n_2 requests. During the time period $\tau_c + (n_2 + 0.5)T_d$, the amount of input data to the client buffer is $Q^2_{in} = \int_t^{t+\tau_c+(n_2+0.5)T_d} E_2[\beta]dt$, and the amount of data consumed by the client from the buffer is $Q^2_{out} =$

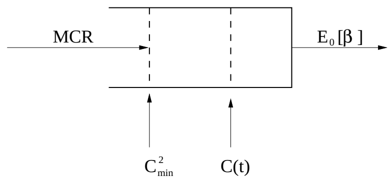

Figure 5: Illustration of C^2_{min} at the client buffer

$\int_t^{t+\tau_c+(n_2+0.5)T_d} E_0[\beta]dt$. Therefore:

$$C(t) + Q^2_{in} - Q^2_{out} \geq 0 \tag{11}$$

C^2_{min} and C_{min} are obtained from the following conditions:

$$C^2_{min} \geq \left(1 - \frac{1}{\rho_{high}}\right)(\tau_c + (n_2+1)T_d)E_0[\beta] \tag{12}$$

$$C_{min} = \max(C^1_{min}, C^2_{min}) \tag{13}$$

It can be seen that the *minimum client buffer fill is directly related to the network congestion, video rate, and the FRTT of the channel*.

3.2 Critical Client Buffer Size

As illustrated in Figure 4, after the FFW/FBW operation is finished, the client will send a signal to the server to restore its outgoing data to normal playback rate. It takes $T_d/2$ for the server to receive the signal and another $T_d/2$ for the client to receive the data sent from the server at the normal playback rate. So, C_f is buffer space required by the client to accumulate this fluctuation.

$$C_f = T_d(\frac{k}{\rho_{high}} - 1)E_0[\beta] \tag{14}$$

From Equations (10), (12), (13) and (14), the critical client buffer size C_c at the client is:

$$C_c = C_f + C_{min} \tag{15}$$

The client buffer overflow probability can be estimated from the critical client buffer size requirement.

4 Results and Discussion

An MPEG video stream with typical values of $X_I = 400$ kbits, $X_P = 200$ kbits and $X_B = 80$ kbits has been used to get the critical client buffer size C_c for different MPEG GoPs. *dino* [4], an MPEG sequence with following parameters has been used in this study. *GoP Pattern:* M3N12 IBBPBBPBBPBB; *Frame Rate:* 24 frames/second; *Quantizer Scale:* 10 for I frame, 14 for P frame and 18 for B frame; *Resolution:* 384*288 pels, 12 bit color information; *Mean Frame Size:* 13078 bits; *Burst Coefficient b:* $\geq$ 9.1; *Peak Bit Rate:* 1.01 Mbps; *Mean Bit Rate:* 0.33 Mbps.

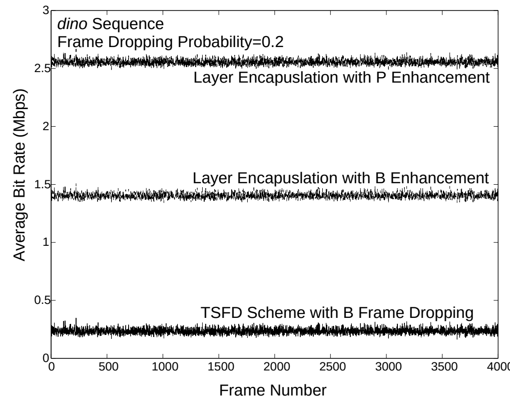

Figure 6: Average bit rate corresponding to three encapsulating scheme for sequence *dino*.

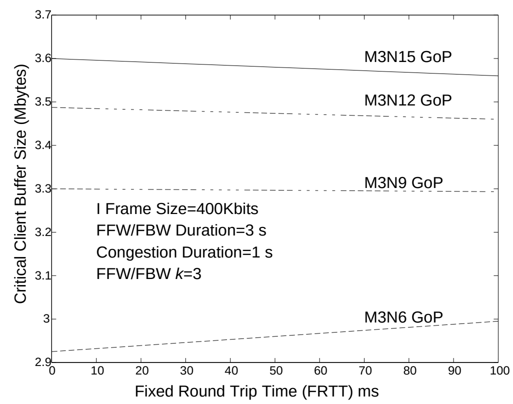

Figure 7: Critical Client buffer size C_c versus fixed round trip time (FRTT).

Figure 6 shows the average bit rate for sequence *dino* corresponding to TSFD scheme with B frame dropping, layer based priority encapsulation with P enhancement, and layer based priority encapsulation with B enhancement. TSFD scheme requires the lowest bandwidth.

The critical client buffer size C_c versus FRTT is shown in Figure 7 for different GoP. As expressed in Equations (10), (12), (13), (14) and (15), the client buffer size depends on FFW/FBW time τ_f, video parameter ρ_{high} and $E_0[\beta]$, and FFW/FBW speed factor k. So, the T_d has different effect on the critical client buffer size for different GoP. High GoP pattern needs a larger buffer size than low GoP pattern since it gets more contribution from B and P frames.

The critical client buffer size C_c versus the FFW/FBW duration with a constant FRTT is shown in Figure 8. As the FFW/FBW time increases, the buffer size increases linearly for all MPEG GoP structures. This is because the longer the FFW/FBW operation, the higher the data consumed by the client.

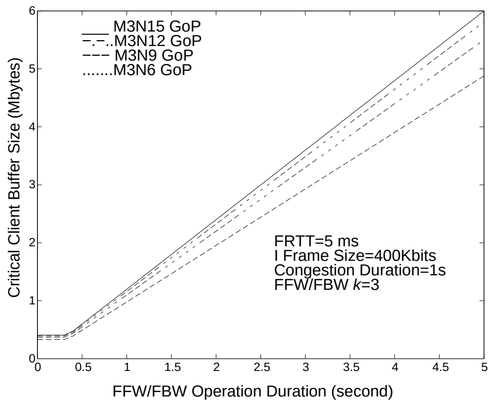

Figure 8: Critical Client buffer size C_c versus FFW/FBW time.

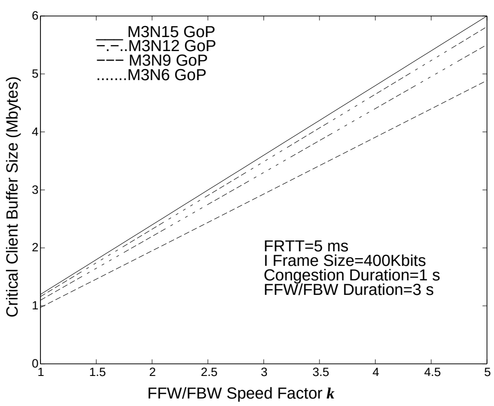

Figure 9: Critical Client buffer size C_c versus FFW/FBW speed factor k.

Higher GoP pattern has a larger ρ_{high} which implies a higher data rate. So, the critical client buffer size for a higher GoP pattern has a larger increasing slope. Note that the fixed part at the beginning is coming from the effect of Equation (14).

Figure 9 shows the critical client buffer size as a function of FFW/FBW speed factor k. As expected from Equation (15), the critical client buffer size increases linearly with the FFW/FBW speed factor. Similarly, because a higher GoP pattern has a larger ρ_{high} and data rate, it also has a larger slope.

Figure 10 shows the critical client buffer size as a function of the network congestion duration. When network congestion lasts for a period of time shorter than nine seconds, the critical client buffer size remains constant. Only when the congestion time become longer than nine seconds, the critical client buffer size increases linearly with the congestion time.

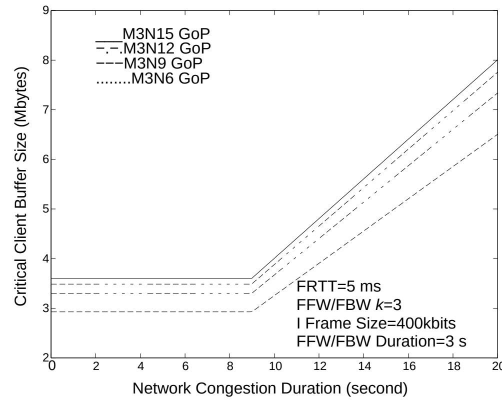

Figure 10: Critical Client buffer size C_c versus network congestion duration.

5 Conclusion

In this paper, we have proposed a Two Stage Frame Dropping (TSFD) for scalable MPEG transmission over an ATM ABR service. Our scheme adaptively sets the value of MCR, and also adjusts the video rate dynamically according to the network congestion. We have also developed a statistical model to determine the client buffer size. We conclude that the client buffer size has a linear relationship with the FFW/FBW operation time and the FFW/FBW speed factor. The client buffer size increases linearly with long term network congestion, and is insensitive to network size, burstiness of the video, and short term network congestion. We have shown that the client buffer requirement depends on the MPEG GoP structure; usually, a large GoP pattern requires a large client buffer. The results in this paper can be used by system and network designers to determine the optimal buffer size and fine tune the network parameters to allow video on demand systems over the ATM ABR service.

References

[1] Ahmed Mehaoua and Raouf Boutaba, "Performance analysis of cell discarding techniques for best effort video communications over ATM networks," *Computer Networks and ISDN Systems*, vol. 29, no. 17-18, pp. 1999–2019, February 1998.

[2] Wenjun Luo and Magda El Zarki, "Quality control for VBR video over ATM networks," *IEEE Journal on Selected areas in Communications*, vol. 15, no. 6, pp. 1029–1039, August 1997.

[3] A. Mehaoua, R. Boutaba, and G. Pujolle, "An extended priority data partition scheme for MPEG video connection over ATM," *IEEE Symposium on Computers and Communication'98*, Alexandria, Egypt, pp. 62–67, July 1998.

[4] O. Rose, "Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM system," research report, Institute of Computer Science, University of Wurzburg, Feb 1995.