Multimedia Over High Speed Networks: Reducing Network Requirements With Fast Buffer Fillup

Bing Zheng and **Mohammed Atiquzzaman**

Department of Electrical and Computer Engineering
The University of Dayton, Dayton, Ohio 45469-0226.
E-mail: *zhengbin@flyernet.udayton.edu, atiq@engr.udayton.edu*

Abstract

The cost effective ABR service is suitable for transmitting bursty compressed video over ATM network. Here, we propose a Fast Buffer Fillup (FBF) scheme to run VoD over ATM network with ABR service. We develop models to determine buffer size at the client and the server for our scheme. We also develop relationships between the startup delay, the ABR service parameters, and the network congestion status. Our results indicate that the new scheme minimizes the startup delay and has simple network resource requirements.

Key Words: *Multimedia transmission, ATM network, ABR service, Buffer size.*

1 Introduction

Because of the high bandwidth and the capability of carrying real-time voice and video, there has been a strong interest in operating Video on Demand (VoD) systems over an ATM network. Available bit rate (ABR) service uses the available bandwidth of the network and controls the data rate of the sources by providing feedback to the source. Compressed video is highly bursty in nature, its bit rate depends on the type of video and also varies significantly between frames. Since ABR has the highest utility of the network resource and offers an acceptable QoS at a low cost, it is important to study the suitability of transmitting *highly interactive* video over ATM using an ABR service.

A VoD system consists of a video source/server, a client including the video decoder/display, and the network over which the video is to be transmitted. The client also need some buffer space to smooth out fluctuations in the data receiving rate from the network. At the start of a session or after a fast forward, the buffer needs to be filled up first before the video can be viewed. This results in a *startup delay* which depends on the buffer size at the client and the bandwidth available from the network. The buffer needs to be proper dimensioned so that the video display is continuous without underflow at the client buffer due to a reduced bandwidth available when the network is congested. Moreover, over-dimensioning the buffer results in expensive client systems and large startup delay.

To increase the network bandwidth efficiency, authors in [1, 2], have investigated the effectiveness of the feedback control scheme in bandwidth allocation and management. They came to a very important conclusion that feedback control is effective in transporting video over an ATM network. Buffer and memory requirements have been discussed in [3]. However, there has been no detailed study on the buffering requirements at the client and server or the network requirements for VoD over the ABR service. A first step towards determining the buffering requirements was been made in [4]. The architecture requires the server to renegotiate bandwidth frequently. In this paper, we have:

- proposed a new *Fast Buffer Fillup* scheme for a VoD system for transporting MPEG-2 video over an ATM network using the ABR service.
- developed analytical models to determine the minimum *server and client buffer sizes* required to allow continuous playback without underflow;
- defined the expression for the *network congestion status* and evaluated the *startup delay* at the client;
- developed the *relationship* between the expected allocated rate, the network congestion status, and the PCR/MCR rate; and

The rest of the paper is organized as follows. In Section 2, we propose the *Fast Buffer Fillup* (FBF) scheme and define the operating principles and the system model. In Section 3, we develop methodologies to determine the minimum buffer requirements for the client and the server. We define and study the startup delay at the client in Section 4. Numerical results are presented in Section 5, with conclusions in Section 6.

2 System Model and Operating Principle

We consider a Video on Demand system (Figure 1) consisting of a video source/server, the ATM back-

0-7803-4984-9/98/$10.00

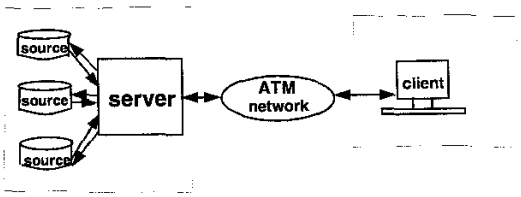

Figure 1: A video on demand system using the ATM network.

bone network and clients with video decoder/display. Video is stored in MPEG-2 compressed format. There are buffers at the client and the server to smooth out fluctuations in the instantaneous data rate of the compressed video and the available bandwidth from the network.

In MPEG-2, video is encoded in three type of frames, namely I, P and B. Frames are arranged in groups of pictures (GoP) denoted by MmNn. An MmNn GoP contains n frames, and the B and P frames are arranged periodically with one P frame and $(m-1)$ frames of B per period. For example, a GoP of M3N9 represents the group IBBPBBPBB. If we denote the data rate of the I, P and B frames by β_I, β_P and β_B respectively, then, the average rate $E[\beta]$ of an MmNn GoP can be expressed as:

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

Next, we outline the assumptions and operating principles of the client, server and the network, which are used to evaluate the performance of our FBF scheme in Sections 3 and 4.

2.1 User model

We assume a *highly interactive* user who can be in one of the four states, viz., Stop, Playback, Fastforward (FFW) and Fastbackward (FBW) corresponding to VCR-like operations. The client responds to the user state by requesting video from source. We assume that during the FFW/FBW operation, the video display scrolls at a very high speed.

2.2 The Client model

- The client buffer must be filled to a minimum fill level $C^C_{\min}$ before the client starts displaying.
- When the client starts from the Stop state, the client only sends a request to server; no other action is required until its buffer reaches $C^C_{\min}$.
- For a FFW/FBW operation, the client sends a FFW/FBW request to the server; at the same time it consumes its buffer at a speed k times that of normal playback.
- At the end of the FFW/FBW operation, the client does not display video until its buffer reaches $C^C_{\min}$.
- The time spent in a FFW/FBW state is much smaller than that spent in the Playback state.

2.3 The server model

- The server negotiates the ABR service parameters with the ATM network during connection setup. These parameters include the PCR (Peak Cell Rate), MCR(Minimum Cell Rate) and ACR (Available Cell Rate). The server transmits data at the PCR (or the maximum bandwidth available from the network) when the client buffer level goes below $C^C_{\min}$. When the buffer level reaches C^C_{min}, the ACR is set to the average rate of the video.
- On receipt of a start or FFW/FBW request, the server requests a bandwidth equal to PCR. If the PCR is not allocated by the network, it accepts whatever bandwidth is offered by the network.
- The server has a buffer to smooth out the data stream sent to the network during the normal playback state.

2.4 Network Performance

The ATM network acts as a transmission channel with Fixed Round Trip Time (FRTT) delay of T_d. In response to the server's request for bandwidth, the network allocates bandwidth to the server with an exponential distribution within the $PCR - MCR$ range(see Eqn. 12).

2.5 Fast Buffer Fill-Up Scheme

When the client moves from the Stop state to the Playback state, or immediately after a FFW/FBW operation, the client buffer needs to be filled to $C^C_{\min}$ before the start of the display. In order to reduce the startup delay, we propose the FBF scheme where the server attempts to renegotiate a bandwidth of PCR from the network in an attempt to fill up the client buffer in the minimum possible time.

3 Client/Server Buffer Requirements

In this section, we model the user behavior and use it to determine the client and server buffering requirements.

3.1 Modeling User Behavior

Figure 2 shows the client which can be in one of the four states: Stop, Playback, Fastforward (FFW) or Fastbackward (FBW). Since the buffer requirement for the client is related to its state, it is necessary to determine the stationary state probability of the client. Let the elements of the state probability vector $V^C = \{V^C_0, V^C_1, V^C_2, V^C_3\}$ represent the Stop, Playback, FFW and FBW states.

We represent the state transitions of the client by a Markov chain as in Figure 3, where $\tau^C_{i,j}$ denotes the state transition probability from state i to state j. By following equation to find the stationary state probabilities.

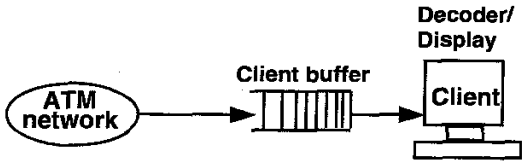

Figure 2: The client model.

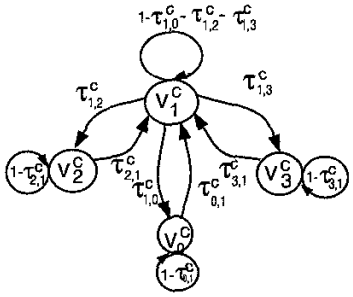

Figure 3: State transitions of the client.

$$V^C = V^C P^C \tag{2}$$

where the state transition matrix P^C for the client is given by:

$$\begin{bmatrix} 1-\tau_{0,1}^C & \tau_{0,1}^C & 0 & 0 \\ \tau_{1,0}^C & 1-\tau_{1,2}^C-\tau_{1,3}^C-\tau_{1,0}^C & \tau_{1,2}^C & \tau_{1,3}^C \\ 0 & \tau_{2,1}^C & 1-\tau_{2,1}^C & 0 \\ 0 & \tau_{3,1}^C & 0 & 1-\tau_{3,1}^C \end{bmatrix}$$

Note that under normal circumstances, the client has a relatively small probability tobe in the FFW/FBW state. From the above transition matrix, we can obtain the steady state probability of the different client states. these equations, we obtain the stationary state probabilities for the client states as:

3.2 Client Buffer Size

At any given time, the expected data rate consumed by the client is given by:

$$E[\lambda] = \sum_{i=0}^{3} V_i^C \lambda_i \tag{3}$$

where λ_i denote the rate at which data is consumed in state i. Since the frame rate of MPEG-2 is constant, the expected data consumption rate $E[\lambda]$ can be expressed as: $E[\lambda] = [V_1^C + k(V_2^C + V_3^C)]E[\beta]$ where, $E[\beta]$ is the average rate of the MPEG-2 video during playback. From our client model in Section 2.2, when the client performs a FFW/FBW operation, the server requests PCR from the network. However, during the bandwidth renegotiation process, the server keeps sending video at the playback rate. The renegotiation involves the server sending an RM cell and waiting for the RM cell to be returned from the

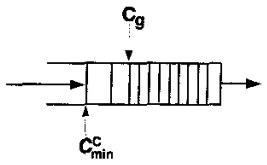

Figure 4: Illustration of C_g and minimum client buffer size ($C_{\min}^C$) for FFW/FBW operation.

ATM network. Because of the fixed round trip delay (FRTT) of the network, the client has to wait for a time $0.5T_d + T_d + 0.5T_d = 2T_d$ before it receives the video at a higher rate. Therefore, the client buffer must have a minimum fill level C_g for no underflow while waiting for the new segment of the video to arrive. During the waiting period, the client consumes video at a rate $k\lambda(t)$ from the client buffer, and the buffer receives video at a rate $\lambda(t)$ from the network. Therefore, the client buffer is depleted at a rate $(k-1)\lambda(t)$ during the waiting period. The minimum buffer size required at the client to prevent underflow during the waiting period of $2T_d$ is therefore given by:

$$C_g = \int_{t_1}^{t_1+2T_d} (k-1)\lambda(t)d(t) \tag{4}$$

where t_1 denotes the time when the client starts a FFW/FBW operation. Since video is sent to the client buffer at the ACR which is approximately equal to the average rate of the video, C_g can be approximately expressed by the expected rate as:

$$C_g = 2(k-1)T_d E[\lambda] \tag{5}$$

However, the client consumes video at a rate which varies from frame to frame, with the I-frame having the highest data rate which is several times higher than the average rate. In the worst case of the user performing the FFW/FBW operation just after the client decodes an I-frame, we must ensure that there is no buffer underflow at the client as shown in Figure 4. To calculate the minimum fill level of the client buffer, we should therefore take the difference between the I-frame rate and the average rate into consideration. Therefore, the minimum fill level $C_{\min}^C$(which is also the minimum capacity of the client buffer) can be expressed as:

$$C_{\min}^C = 2(k-1)T_d E[\lambda] + T_f(E[\beta_I] - E[\beta]) \tag{6}$$

where $E[\beta_I]$ and $E[\beta]$ are the average rates of the I-frame and the GoP respectively. T_f is the time duration for a single frame of MPEG-2 video, and $T_f(E[\beta_I] - E[\beta])$ accounts for the buffer required due to the rate difference between the I-frame rate and the average rate. From Eqn.(6), the minimum client buffer size depends on T_d, $E[\beta]$ and $E[\lambda]$. $E[\beta]$ depends on the type of the movie, and $E[\lambda]$ depends on

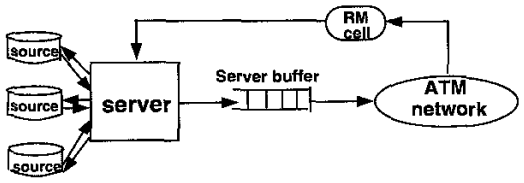

Figure 5: The server model

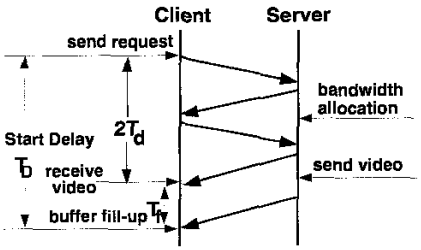

Figure 6: Illustration of startup delay after a trick mode.

$E[\beta]$ and the level of interactivity of the user. Since we can assume T_d to be a constant for a network, the client buffer size is therefore determined by the type of movie and the level of user interactivity.

3.3 Server Buffer Size

Since the server sends video at the ACR denoted by μ_a, for no overflow/underflow at the server buffer, the long term dynamic variation of the server buffer accumulation per GoP should be zero:

$$\sum_{\text{all GoP}} \delta(\text{buffer accumulation in a GoP}) = 0 \quad (7)$$

The expected value of ACR ($E[\mu_a]$) should therefore satisfy the following condition:

$$\sum_{\text{all GoP}} E[\mu_a] n T_f = \sum_{\text{all GoP}} \beta_I T_f + \beta_P (n/m - 1) T_f + \beta_B (m-1) n/m T_f \quad (8)$$

The left hand side of Eqn.(8) represents the amount of data sent by the server, and the right hand side represents the sum of the data contained in movie. Substituting Eqn.(1) in Eqn.(8), we get:

$$E[\mu_a] = E[\beta] \quad (9)$$

Because of the difference in ACR and the I-frame rate, the server buffer can be determined by taking into account the fact that difference must be stored in the server buffer. Therefore, the minimum server buffer capacity $C^s_{\min}$can be expressed as:

$$C^s_{\min} = (E[\beta_I] - E[\beta]) T_f \quad (10)$$

Typically, for an MPEG-2 video with an M3N9 GoP, $\beta_I = 8.25$ Mb/s, $\beta_P = 2.25$ Mb/s and $\beta_B = 0.6$ Mb/s [5]. For a 30 frames per second video, $T_f = 0.033$ sec and Eqn.(1) gives $E[\beta] = 1.817$ Mb/s. From Eqn.(10), we obtain the minimum server buffer size to be about 25 Kbytes.

4 Startup Delay Characteristics

We define the *startup delay* as the time between the user presses Playback (from the Stop state) to the start of the video display, or the time between the user presses FFW/FBW to the start of the video. The startup delay (T_D) consists of two parts: a fixed part arising due to the FRTT of the network and a dynamic part T_f which is required to fill the client buffer to the minimum fill level $C^C_{\min}$ as shown in Figure 6.

$$T_D = 2T_d + T_f \quad (11)$$

Assuming T_d as constant, the startup delay depends on the minimum buffer fill level and the fillup rate which depends on the bandwidth allocated by the network. When the server requests PCR from the network, the network may not be able to allocate the requested bandwidth. We assume an exponential distribution for the ACR and we can express the approval probability of an ACR of μ_r by:

$$P(\mu_r) = e^{-\alpha \frac{\mu_r - \mu_m}{\mu_p - \mu_m}} \quad (12)$$

where α is a parameter describing the network congestion; μ_p and μ_m are the PCR and MCR respectively. We can obtain the expected value of the ACR ($E[\mu_r]$) during FFW/FBW as:

$$E[\mu_r] = \frac{\int_{\mu_m}^{\mu_p} \mu_r P(\mu_r) d\mu_r}{\int_{\mu_m}^{\mu_p} P(\mu_r) d\mu_r} \quad (13)$$

The denominator in Eqn. (13) can be expressed as:

$$\int_{\mu_m}^{\mu_p} P(\mu_r) d\mu_r = \frac{e^{-\alpha} - 1}{a} \quad (14)$$

where $a = -\frac{\alpha}{\mu_p - \mu_m}$. The numerator in Eqn.(13) can be expressed as:

$$\int_{\mu_m}^{\mu_p} \mu_r P(\mu_r) d\mu_r = \frac{1}{a^2}(a\mu_p e^{a(\mu_p - \mu_m)} - e^{a(\mu_p - \mu_m)} - a\mu_m + 1) \quad (15)$$

Substituting Eqns.(14) and (15) into Eqn.(13) we get

$$E[\mu_r] = \left(1/\alpha + \frac{1 - qe^{-\alpha}}{(q-1)(1-e^{-\alpha})}\right)(q-1)\mu_m \quad (16)$$

where $q = \mu_p/\mu_m$ is the ratio of the PCR to MCR. To study the relationship between the expected ACR and the PCR, we differentiate Eqn.(16) with respect to q to obtain:

$$\frac{\partial E[\mu_r]}{\partial q} = \left(\frac{1}{\alpha} - \frac{e^{-\alpha}}{1 - e^{-\alpha}}\right)\mu_m \quad (17)$$

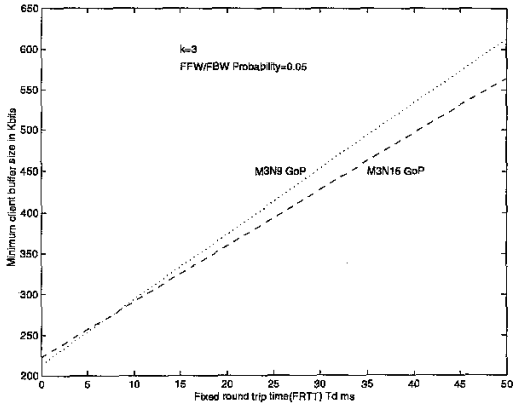

Figure 7: The minimum client buffer size versus FRTT for FFW/FBW probability of 0.05.

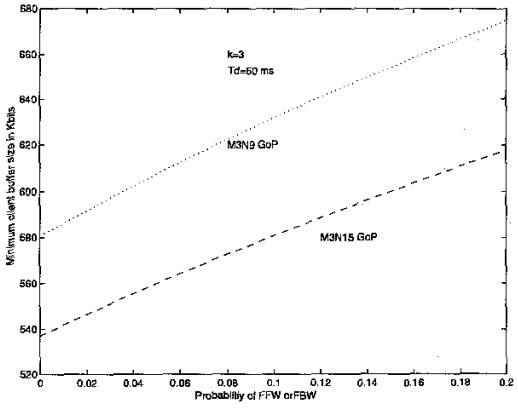

Figure 8: The minimum client buffer size versus FFW/FBW probability for $T_d = 50$ ms.

From Eqn.(16), we see that $E[\mu_r]$ contains q. However, from Eqn.(17), $\frac{\partial E[\mu_r]}{\partial q}$ is independent of q. Therefore, we conclude that $E[\mu_r]$ linearly depends on q. Since MCR is chosen equal to the average rate of the video which is constant for a specific movie, $E[\mu_r]$ therefore directly depends on PCR. Moreover, since the buffer fillup rate directly depends on $E[\mu_r]$, we conclude that the expected buffer fillup rate varies linearly with PCR. This implies that the startup delay can be minimized simply by requesting a high PCR during connection set up. The expected dynamic part of the startup delay $E[T_f]$ in Eqn.(11) can be obtained by:

$$E[T_f] = \int_{\mu_m}^{\mu_p} \frac{C_{\min}^{C}}{\mu_r} P(\mu_r) d\mu_r \qquad (18)$$

Since there is no closed form solution for the above integration, we evaluate it by numerical integration.

5 Numerical Results

The relationship between the minimum client buffer size and the FRTT as shown in Figure 7 for a probability of 0.05 of moving to the FFW/FBW state, and two different GoPs representing two different video rates. We notice that the minimum client buffer size increases linearly with the FRTT. Note that an M3N9 GoP has a higher bit rate than an M3N15 GoP. For large values of FRTT (corresponding to a WAN/MAN), the buffer required for M3N9 GoP is larger than for M3N15 GoP. For small FRTT (corresponding to a LAN), M3N15 GoP requires a larger buffer than M3N9 GoP. This is explained by the fact that for small FRTT, the first part of Eqn. (6) is small and the client buffer size is dominated by the second term of Eqn. (6) which is the rate difference between the I-frame and the average rate.

Figure 8 illustrates the minimum client buffer size corresponding to FFW/FBW probabilities in the range 0-0.2 with a fixed FRTT. Since an M3N9 GoP has a higher data rate than an M3N15 GoP, M3N9 GoP requires a larger buffer.

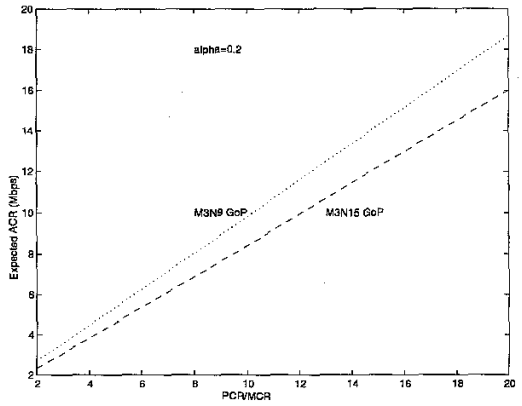

Figure 9: Expected ACR versus PCR/MCR for a constant network congestion status.

In FBF scheme, the server requests a high bandwidth only if the client performs a FFW/FBW operation or performs a Start operation. It doesn't need to renegotiate bandwidth for each GoP as in [4]. Therefore, the FBF scheme simplifies bandwidth allocation for the network at the price of increasing the minimum buffer size requirement at the client.

The expected ACR (obtained from Eqn. (16)) versus the PCR for a constant network congestion is shown in Figure 9, while the expected ACR versus the network congestion status for a constant PCR/MCR ratio is shown in Figure 10. The expected ACR depends on the network congestion status and the negotiated PCR and MCR during connection set up. From Figure 9, we find that the expected ACR increases linearly with an increase in the ratio of

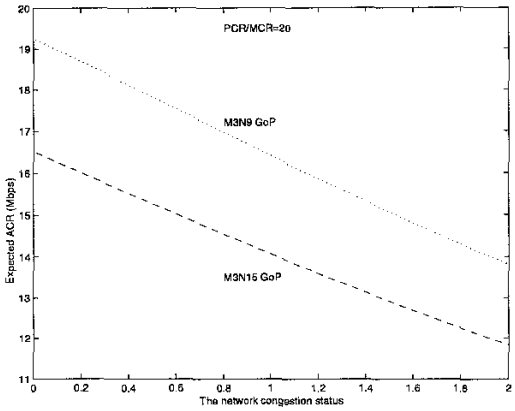

Figure 10: Expected ACR versus the network congestion status for fixed PCR/MCR.

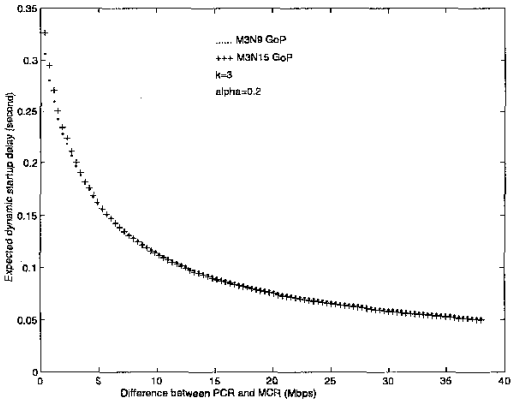

Figure 11: Expected startup delay versus PCR.

PCR/MCR as predicted Eqn. (16). This implies that the expected ACR can be set to a desired value by setting a suitable PCR during connection set up.

Figure 10 shows that the expected ACR decrease rapidly when the network congestion increases. This means that when the network is congested, the startup delay will increase dramatically. Also, Figures 9 and Figure 10 show that the expected ACR depends only on the network congestion status.

The expected value of the dynamical startup delay ($E[T_f]$) versus the difference $PCR - MCR$ is calculated from Eqn. (18) and is shown in Figure 11. We find that the M3N9 and M3N15 GoPs have the same dynamic startup delay for the same level of network congestion. Moreover, ($E[T_f]$) decreases exponentially with higher values of PCR (MCR is chosen to equal to the average rates of the video).

6 Conclusion

In this paper, we proposed the FBF scheme to run video on demand over the ABR service of an ATM network. We have developed models for the client, the server, the network and a highly interactive user. We have used the models to determine the minimum buffering requirements at the client and the server. The proposed FBF scheme simplifies the networking bandwidth allocation because it does not need to renegotiate bandwidth frequently.

Numerical results show that the minimum size of the client buffer depends linearly on the FRTT of the system. It also depends on the GoP which affects the burstiness of the video stream. For a LAN, the contribution to the minimum client buffer size is dominated by the burstiness of the video stream. For a WAN, the minimum client buffer size is mainly decided by the $FRTT$ of the network. The startup delay depends directly on the network congestion status, the PCR, and MCR negotiated at connection set up. The higher the PCR, the faster is the filling up of the client buffer. Results also illustrate that there is no difference in the startup delay for different GoPs. Therefore, a video with any GoP can be used as the source in a multimedia system using the proposed FBF scheme.

References

[1] Hemant Kanakia, Partho P. Mishra, and Amy R. Reibman, "An adaptive congestion control scheme for real time packet video transport," *IEEE/ACM Transaction on Networking*, vol. 3, no. 6, pp. 671–682, December, 1995.

[2] Marwan Krunz and Satish K. Tripath, "Exploiting the temporal structure of MPEG-2 video for the reduction of bandwidth requirement," *IEEE INFOCOM'97*, Kobe, Japan, pp. 143–150, April 1997.

[3] Jean M. Macmanus and Keith W. Ross, "Video on demand over ATM: constant-rate transmission and transport," *Proceedings of INFOCOM'96*, San Francisco, pp. 1357–1362, March 1996.

[4] Bing Zheng and Mohammed Atiquzzaman, "Video on demand over ATM: system design and networking requirements," *ENCOM'98-The Enterprise Networking and Computing'98*, Atlanta, June 7-11 1998.

[5] Lawrence G. Roberts, "Can ABR service replace VBR service in ATM network," *Proceedings of the COMPCON'95 Conference*, Piscatway, New Jersy, pp. 346–348, 1995.