Renegotiated CBR Transmission in Interactive Video-on-Demand System

Noriaki Kamiyama, Victor O.K. Li

Communication Sciences Institute
University of Southern California
Los Angeles, CA 90089-2565
Email:{noriaki,vli}@irving.usc.edu

Abstract

An Interactive Video-On-Demand (IVOD) system requires transmission bandwidth allocation for each user. Since the volume of data in each video frame is variable, dynamic bandwidth allocation is desirable. In this paper, a new scheme that dynamically determines required bandwidth based on the queue length at the viewers Set-Top Box (STB) is proposed. This method requires no pre-calculation, so it is easily applied to IVOD. The variance of the video transmission rate for each user is an important factor as it affects the service quality of other multiplexed traffic. It is desirable that the transmission rate is changed gradually. A multi-layer concept is introduced to achieve this. Through numerical evaluation using actual movie data, we demonstrate that the variance of the transmission rate is close to the optimal value and the bandwidth utilization is close to unity.

1 Introduction

A Video-On-Demand (VOD) system enables a viewer to choose a video program from a large selection without leaving his home. It can offer a wide range of services. One extreme is Near VOD (NVOD) service in which a point-to-multipoint stream is generated and multiple viewers are serviced by one stream to reduce the system cost at the expense of service delay (the viewer must request playback between five and 30 minutes prior to the start of playback[1]). The other extreme is Interactive VOD (IVOD) services in which a stream is provided to each individual viewer with full interactivity. Since the viewer can use the stream privately, the video begins to play upon request without delay. VCR-like controls, i.e. pause, fast forward, rewind, etc. are also available.

The VOD system consists of a video server, a backbone network, an access network, and a Set-Top Box (STB)[2]. The video server stores encoded video and retrieves it upon request. The retrieved data is transmitted over the backbone and access network to the STB in the viewer's home. The main functions of STB are to absorb delay jitter and to decode video frames. Video is encoded into digital data and stored in the video server. The encoder can be classified into two categories: variable rate and constant rate. The variable rate encoder provides constant video quality, while the constant rate encoder gives variable quality services. Although variable quality services can be provided at a lower cost, the quality degradation is severe in frames whose redundancy are small (such as a scene change). Therefore, constant quality service is preferable. A popular encoding scheme, MPEG, belongs to this category[3]. *In this paper, we only consider the IVOD system which provides constant quality service.*

Since most existing networks provide constant bit rate (CBR) service, the accommodation of variable bit rate (VBR) video on CBR network has received much attention in the literature. The challenge is due to the bursty nature of VBR video and its delay sensitivity. The delay sensitivity makes it difficult to apply closed-loop congestion control schemes. So open-loop congestion control is necessary for VBR video service. However, this control is also difficult because of the bursty nature of VBR video. To allocate bandwidth, the traffic parameters must be specified for Call Admission Control (CAC). Moreover, specified traffic parameters must be enforced at each Network User Interface (NUI) in a packet-switched network (this control is called Usage Parameter Control (UPC)). This traffic specification is quite difficult for VBR video. Moreover, it seems difficult to obtain a statistical multiplexing gain by multiplexing many VBR video sources because they have a self-similar property[4].

Since CAC and UPC are simple for CBR traffic, one approach [5],[6] is to treat VBR video as CBR. Although this resolves the above problems, large pre-loading delay[†] and memory size are required to fit VBR traffic with a constant allocated bandwidth. Another solution is to renegotiate the CBR rate during the session, thereby decreasing the required pre-loading delay and STB memory[7],[8],[9]. However, the complex pre-calculation and associated delay necessary to obtain a bandwidth sequence table makes the scheme unsuitable for a real-time application such as IVOD. Whenever a viewer makes an interactive operation, he must wait for the pre-calculation.

In this paper, we propose a new dynamic bandwidth allocation method. Although this method is in the renegotiated CBR category, it needs no pre-calculation. Therefore, IVOD with constant quality service is easily realized. To avoid both overflow and underflow at STB, the allocated bandwidth is determined based on the queue length at STB. The Coefficient of Variation (CoV) of the transmission rate (allocated bandwidth) should be as small as possible to minimize the impact on other traffic. Hence, we introduce a multiple layer approach to deal with the fluctuation of the STB queue length. Using real data from the MPEG encoded movie *Star Wars*, it is demonstrated that

[†]The pre-loading delay is the time difference between the arrival of the first data frame at the STB and the beginning of the playback. This delay is introduced to avoid STB buffer underflow.

0-8186-7819-4/97 $10.00 © 1997 IEEE

the CoV of the transmission rate is close to the ideal value and the network utilization is close to unity.

The remainder of this paper is organized as follows. In Section 2, we review previously proposed bandwidth allocation methods. In Section 3, we define the proposed scheme. We introduce the multi-layer approach in Section 4. In Section 5, we investigate the performance of the new method using actual MPEG traces. Finally, in Section 6, we conclude our study.

2 Bandwidth Allocation in VOD

In this section, existing bandwidth allocation methods proposed for stored video transmission is briefly summarized.

Video playback requires continuous data arrival. If the STB buffer is empty at a new frame playback time, the video playback is interrupted. New arriving data is discarded when the STB buffer is full; this degrades the quality. Therefore, a bandwidth allocation method is required to avoid both buffer underflow and overflow at STB.

CBR and Renegotiated CBR (RCBR)[5],[6],[8] bandwidth allocation methods can be classified into two major categories: (1) fixed allocation (CBR) and (2) dynamic allocation (RCBR). Moreover, these methods are distinguished by the frame period treatment in the network (see Fig. 1). One maintains a frame period, and the other transmits video data without considering a frame period.

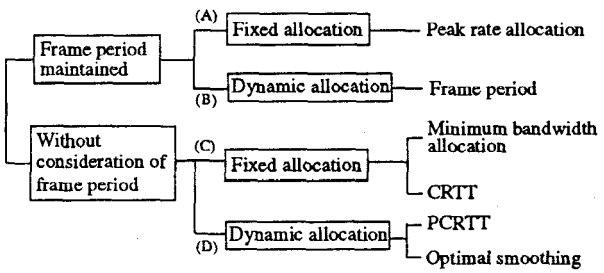

Figure 1: CBR accommodation methods on VOD

In the former, the video data forming one frame arrives at STB during one frame period, so STB buffer equal to one maximum frame size is enough to avoid both overflow and underflow. If the CBR bandwidth is allocated only at the call set-up time ((A) in Fig. 1), the peak rate must be used. Although this allocation is the simplest, network utilization is low. Dynamic bandwidth allocation with frame period preservation ((B) in Fig. 1) can be also considered. This requires bandwidth renegotiation at each frame boundary, and is proposed for real-time video transmission, such as video conferencing[10]. Although this method improves the network utilization, bandwidth renegotiation is very frequent. The network load will increase because of renegotiation signalling traffic. The other problem with frame period preservation is that the variation of the transmission rate tends to be large. This affects QOS of other multiplexed traffic.

Therefore, video transmission without considering a frame period is superior. In this case, the key is how to avoid buffer underflow and overflow. Next, some existing methods are reviewed.

2.1 Fixed allocation

In [6], the minimum bandwidth R^* which avoids buffer underflow is allocated. By increasing the STB buffer size and the pre-loading delay, R^* can be decreased. The actual transmission rate is controlled to be smaller than or equal to R^* to avoid buffer overflow. When an interactive operation changes the frame sequence, R^* must be re-calculated. Since this calculation takes time, it is difficult to apply this method to IVOD. Another problem is that the pre-loading delay may be very large. For example, it is $37s$ for *Star Wars* when the STB memory size is $16Mbytes$.

The Constant-Rate Transmission and Transport (CRTT) method[5] pre-calculates the allocated bandwidth avoiding overflow as well as underflow at the STB buffer. The actual transmission rate is always constant (equal to the allocated bandwidth). Although this method is easy to manage, it is also difficult to apply it to IVOD because of its complex pre-calculation. Moreover, the required STB buffer size and the pre-loading delay are large (in the case of *Star Wars*, $22.4Mbytes$ memory and $37s$ pre-loading are required).

2.2 Dynamic allocation

If the frame period is preserved in the network, the bandwidth is renegotiated at every frame period. By ignoring the frame period, the renegotiation period can be made much longer. Piecewise Constant Rate Transmission and Transport (PCRTT)[5] is an improved CRTT. By allowing the allocated bandwidth to vary, the required STB buffer size and pre-loading delay can be decreased. The problem is how to determine the scheduling table of the transmission rate. To minimize the impact to other traffic, the transmission rate sequence should be as smooth as possible (i.e. small CoV of transmission rate). The ideal rate sequence is pure-CBR (CRTT realizes this ideal case). The optimal smoothing method[8] tries to find the transmission rate schedule minimizing the CoV and the peak transmission rate. This method, however, requires a complex pre-calculation. In the case of *Star Wars*, it takes around $8s$ when an R4400 ($150MHz$) CPU is used. Thus, this scheme can not be used in IVOD.

3 RCBR based on STB Queue

All existing methods mentioned above need a complex pre-calculation to obtain the fixed bandwidth or the transmission rate schedule. As a result, it is difficult to apply them in IVOD. Now we propose a new dynamic bandwidth allocation scheme in which no pre-calculation is necessary.

3.1 Basic concept

In order to avoid both underflow and overflow at the STB buffer, the transmission rate and its timing of renegotiation are determined dynamically based on the STB queue length. In this paper, we assume that the backbone and access network is a packet-switched network. Since the routing procedure and the bandwidth allocation can be separated, quick bandwidth renegotiation is possible. For example, bandwidth can be renegotiated using Fast Reservation Protocol (FRP) in the case of ATM[11]. When STB wants to change the allocated bandwidth, it sends a control packet including new bandwidth information to the server. Each intermediate node on the path compares it with the link capacity available. The video server sends back an acknowledgment or reject packet to STB. If at least one intermediate link can not accommodate the requested bandwidth, this FRP is rejected†. Therefore, it takes one round-trip time to complete the bandwidth renegotiation. Hereafter, we use the term "FRP" to represent the process of bandwidth renegotiation.

The basic idea of the proposed method, called Dynamic bandwidth Allocation based on Queue length at STB (DAQS), is to increase the bandwidth when a buffer underflow is predicted and to decrease it when a buffer overflow is predicted. The important thing is to determine when the bandwidth should be renegotiated and what the new bandwidth should be.

3.2 Formulation

The time axis is divided into discrete units equal to the video frame time. This time index represents the frame number played-back at STB plus the pre-loading time d. At the beginning of each frame playback, all information bits in the frame are assumed to be removed from the STB buffer instantaneously. The following random variables and parameters are defined (see Fig. 2).

- x_n $(bits)$: Volume of stored data in STB at $t = n$ just **before** one frame is removed
- x_n^+ $(bits)$: Volume of stored data in STB at $t = n$ just **after** one frame is removed
- y_n $(bits)$: Size of the frame played-back at $t = n$
- r_n $(bits/s)$: CBR rate between $t = n$ and $t = n + 1$
- r_I $(bits/s)$: Initial transmission rate
- F $(fps(frame\ per\ sec.))$: Reciprocal of frame time length (i.e. frame rate)
- d : Pre-loading time in frame length (first frame is played-back at $t = d$)
- B $(bits)$: STB buffer size
- ω : FRP control delay
- Y_{max} $(bits)$: Maximum frame size
- Y_{min} $(bits)$: Minimum frame size
- Y_{av} $(bits)$: Average frame size
- V : Number of frames in one movie

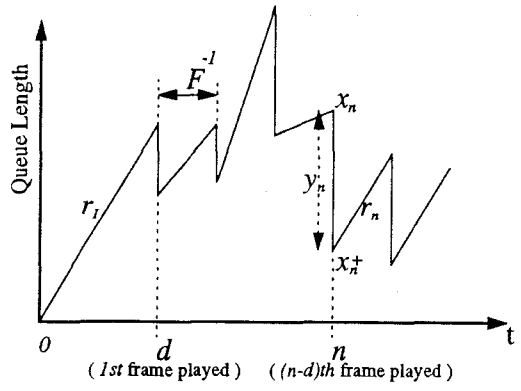

Figure 2: Definition of random variables

If the round-trip time is T, ω is given by $\omega = \lceil F \cdot T \rceil$ where $\lceil x \rceil$ stands for the minimum integer which is more than or equal to x. It is noted that the unit of ω is the video frame length. Bandwidth renegotiation takes ω frames, so a change of the transmission rate $r_{n+\omega}$ must be initiated at $t = n$. Therefore, STB must predict the queue length $x_{n+\omega+1}$ and $x^+_{n+\omega+1}$ at $t = n$ to avoid both underflow and overflow at the STB buffer. From Fig. 2,

$$\begin{cases} x_n &= x^+_{n-1} + \frac{r_{n-1}}{F}, \\ x^+_n &= x_n - y_n \end{cases} \tag{1}$$

are obtained. In order to avoid both underflow and overflow at the STB buffer, $x^+_n \geq 0$ and $x_n \leq B$ are required for all n. It is assumed that bandwidth is not renegotiated before the playback starts ($t < d$). So r_n can be controlled only when $n \geq d + \omega$. We consider the restrictions for r_n $(n \geq d + \omega)$. From (1), the following is derived:

$$x^+_{n+1} = x^+_{n-\omega} + \frac{\sum_{i=n-\omega}^{n} r_i}{F} - \sum_{j=n-\omega+1}^{n+1} y_j.$$

As $x^+_{n+1} \geq 0$ is required, it is necessary that

$$r_n \geq F \sum_{j=n-\omega+1}^{n+1} y_j - F x^+_{n-\omega} - \sum_{i=n-\omega}^{n-1} r_i.$$

Considering the maximum value of the right side leads to

$$r_n \geq (\omega + 1) F Y_{max} - F x^+_{n-\omega} - \sum_{i=n-\omega}^{n-1} r_i. \tag{2}$$

In the same way, the following is obtained,

$$x_{n+1} = x^+_{n-\omega} + \frac{\sum_{i=n-\omega}^{n} r_i}{F} - \sum_{j=n-\omega+1}^{n} y_j.$$

†For all dynamic bandwidth allocation schemes, there is a possibility the network may reject request for additional bandwidth. In this case, we can just discard some of the frames.

As $x_{n+1} \leq B$ is required, we have

$$r_n \leq FB + F\sum_{j=n-\omega+1}^{n} y_j - Fx_{n-\omega}^{+} - \sum_{i=n-\omega}^{n-1} r_i.$$

Considering the minimum value of the right side leads to

$$r_n \leq FB + \omega FY_{min} - Fx_{n-\omega}^{+} - \sum_{i=n-\omega}^{n-1} r_i. \quad (3)$$

Replacing n with $n+\omega$ in (2) and (3), we obtain the inequality which r_n must satisfy for $d \leq n \leq V-\omega$,

$$(\omega+1)FY_{max} - f_n(x^+, r) \leq r_{n+\omega} \leq FB + \omega FY_{min} - f_n(x^+, r) \quad (4)$$

where

$$f_n(x^+, r) \equiv Fx_n^+ + \sum_{i=n}^{n+\omega-1} r_i.$$

For r_n to satisfy (4),

$$B \geq (\omega+1)Y_{max} - \omega Y_{min}. \quad (5)$$

Solving (4) for x_n^+, and setting $r_{n+\omega} = r_{n+\omega-1}$, we have

$$(\omega+1)Y_{max} - g_n(r) \leq x_n^+ \leq B + \omega Y_{min} - g_n(r), \quad (6)$$

where

$$g_n(r) \equiv \frac{2r_{n+\omega-1}}{F} + \sum_{i=n}^{n+\omega-2} \frac{r_i}{F}.$$

So long as (6) is satisfied at $t=n$, there is no need to change the transmission rate at $t=n+\omega$, i.e. $r_{n+\omega} = r_{n+\omega-1}$, and FRP at $t=n$ is not necessary. However FRP at $t=n$ is required when (6) is not satisfied. If the bandwidth must be increased, i.e. $x_n^+ < (\omega+1)Y_{max} - g_n(r)$, the new bandwidth at $t=n+\omega$ can be set as

$$r_{n+\omega} = (\omega+1)FY_{max} - f_n(x, r), \quad (7)$$

considering (4). If the bandwidth must be decreased, i.e. $x_n^+ > B + \omega Y_{min} - g_n(r)$, $r_{n+\omega}$ can be set as

$$r_{n+\omega} = FB + \omega FY_{min} - f_n(x, r). \quad (8)$$

As mentioned before, the transmission rate is kept constant before $t = d+\omega$, i.e.

$$r_0 = r_1 = \cdots = r_{d+\omega-1} \equiv r_I.$$

We must consider the restrictions for d and r_I to prevent both overflow and underflow before $t = d+\omega$. From (1), we obtain

$$\begin{cases} max(x_n) &= max(x_{n-1}) + \frac{r_I}{F} - Y_{min}, \\ min(x_n^+) &= min(x_{n-1}^+) + \frac{r_I}{F} - Y_{max} \end{cases}$$

for $0 \leq n \leq d+\omega$. Assuming that $FY_{min} \leq r_I \leq FY_{max}$, we have

$$\begin{cases} max(x_n) \geq max(x_{n-1}), \\ min(x_n^+) \leq min(x_{n-1}^+) \end{cases}$$

for $d \leq n \leq d+\omega$. As $y_n = 0$ when $0 \leq n \leq d-1$, we obtain

$$\begin{cases} max(x_n) > max(x_{n-1}), \\ min(x_n^+) > min(x_{n-1}^+) \end{cases}$$

for $1 \leq n \leq d-1$. Therefore, we only have to consider $max(x_{d+\omega}) \leq B$ and $min(x_{d+\omega}^+) \geq 0$. From (2) and (3), the following inequalities:

$$\begin{aligned} \frac{r_I}{F}(d-1) + (\omega+1)\frac{r_I}{F} - \omega Y_{min} &\leq B \\ \frac{r_I}{F}(d-1) + (\omega+1)\left(\frac{r_I}{F} - Y_{max}\right) &\geq 0 \end{aligned}$$

are obtained. Thus the restriction for d and r_I is derived as

$$\begin{cases} (\omega+1)FY_{max} \leq (d+\omega)r_I \leq FB + \omega FY_{min}, \\ FY_{min} \leq r_I \leq FY_{max}. \end{cases} \quad (9)$$

The pre-loading time length d should be as small as possible, so we set r_I as the maximum allowed value within $FY_{min} \leq r_I \leq FY_{max}$ at each call set-up time. It means that $r_I = min(FY_{max}, E)$ where E is the minimum available link capacity on the route. Moreover d is derived as

$$d = \lceil \frac{(\omega+1)FY_{max}}{r_I} \rceil - \omega.$$

For d and r_I to exist, the restriction

$$B \geq (\omega+1)Y_{max} - \omega Y_{min}$$

is necessary. This inequality is the same as (5).

Now, we have determined the bandwidth requirements which avoid both underflow and overflow at the STB buffer. Next, we describe a simple protocol named DAQS-Simplest Protocol (DAQS-SP) to implement these requirements. It is described as follows.

[DAQS-SP]

Line 1 Transmission starts at $t=0$ with a fixed rate of r_I. Here, r_I satisfies $FY_{min} \leq r_I \leq FY_{max}$ and takes the largest available value.

Line 2 Playback starts at $t=d$. Here, $d = \lceil \frac{(\omega+1)FY_{max}}{r_I} \rceil - \omega$.

Line 3 At the instance when one frame is removed from the buffer at $t=n$ ($d \leq n \leq V-\omega$), the Algorithm Module 1 (AM1) depicted in Fig. 3 is executed.

DAQS-SP guarantees no buffer underflow and overflow so long as FRP is accepted. However, bandwidth is renegotiated only when the STB queue length becomes too short (the buffer underflow is predicted) or too large (the buffer overflow is predicted). Therefore, the allocated bandwidth almost always takes two values: a big one (close to FY_{max}) and a small one (close to FY_{min}). This means the transmission rate will oscillate between two extreme values. In order to decrease the impact to other traffic, the transmission rate should be as smooth as possible. Thus it is a good idea to change the bandwidth gradually. In the next section, a multi-layer concept realizing this modification is proposed.

```
begin
  if x_n < (w+1)Y_max - g_n(r) then
  begin
    r_{n+w} = (w+1)FY_max - f_n(x+, r)
    Start FRP
  end
  else if x_n > B+wY_min - g_n(r) then
    begin
      r_{n+w} = FB+wFY_min - f_n(x+, r)
      Start FRP
    end
    else Set r_{n+w} = r_{n+w-1}
end.
```

Figure 3: Algorithm module 1

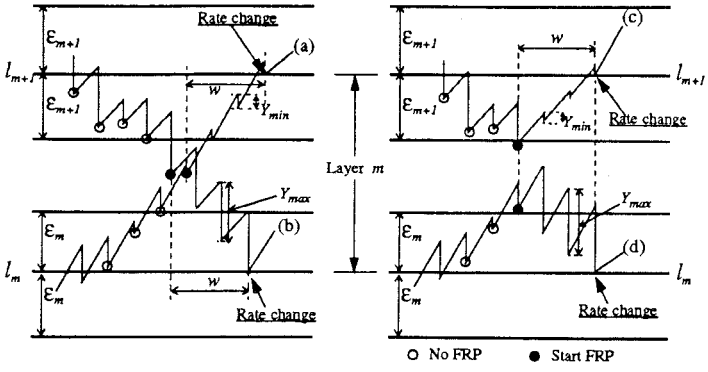

Figure 4: Rate transition in middle layers

4 Introduction of Multi-Layer

The STB buffer is divided into M ($M \geq 3$) layers. DAQS-SP corresponds to the case of $M = 2$. Let l_m ($1 \leq m \leq M-1$) denote the boundary between layer $m-1$ and layer m. In layer 0, the queue length is checked at each frame period based on AM1 to avoid buffer underflow. In layer $M-1$, the queue length is also checked at each frame period based on AM1 to avoid buffer overflow. As a result, the rate is changed dynamically in layer 0 and $M-1$. On the other hand, the volume of video data arriving at STB within a frame period is fixed at Q_m for layer m ($1 \leq m \leq M-2$). If Q_m is increased gradually with decreasing m, the degree of bandwidth increment and decrement becomes small. So a decrease in the CoV of the transmission rate is expected. Since the minimum frame size is Y_{min} and the maximum frame size is Y_{max}, we should set Q_m as

$$Y_{min} < Q_{M-2} < Q_{M-3} < \cdots < Q_2 < Q_1 < Y_{max}. \quad (10)$$

If the queue length oscillates around a layer boundary, FRP is used in almost every frame. In order to increase the FRP interval, a parameter ϵ_m ($1 \leq m \leq M-1$) is introduced. If the queue length increases from layer $m-1$ to layer m, an FRP changing the transmission rate to FQ_m is requested only when the queue length exceeds $l_m+\epsilon_m$ (see Fig. 4). In the same way, if the queue length decreases from layer $m+1$ to layer m, an FRP changing the transmission rate to FQ_m is requested only when the queue length falls bellow $l_{m+1} - \epsilon_{m+1}$.

4.1 Setting ϵ_m

Considering the definition of Q_m and l_m, the allocated bandwidth renegotiation from FQ_{m-1} or FQ_{m+1} to FQ_m must be completed when x_n^+ is in layer m (i.e. $l_m \leq x_n^+ \leq l_{m+1}$). From the worst case ((a) of Fig. 4), the following restriction for l_m ($m = 1, 2, \cdots, M-2$) is obtained:

$$l_{m+1} - l_m \geq \epsilon_m + (\omega+1)\left(Q_{m-1} - Y_{min}\right).$$

Let Q_0 denote the maximum transmitted data within one frame in layer 0, i.e. Y_{max}. Moreover, the worst case ((b) of Fig. 4) gives the following restriction for $m = 1, 2, \cdots, M-2$:

$$l_{m+1} - l_m \geq \epsilon_{m+1} + (\omega+1)\left(Y_{max} - Q_{m+1}\right).$$

Let Q_{M-1} denote the minimum transmitted data within one frame in layer $M-1$, i.e. Y_{min}. As a result, the restriction for l_m ($m = 1, 2, \cdots, M-2$) is represented as

$$\begin{aligned} l_{m+1} \quad \geq \quad & l_m + max\left\{\epsilon_m + (\omega+1)(Q_{m-1} - Y_{min}),\right. \\ & \left.\epsilon_{m+1} + (\omega+1)(Y_{max} - Q_{m+1})\right\}. \end{aligned} \quad (11)$$

The worst cases (c) and (d) of Fig. 4 give the restriction

$$\epsilon_m \geq max\left\{\omega(Y_{max} - Q_{m-1}) - 1,\ \omega(Q_m - Y_{min}) - 1\right\}, \quad (12)$$

for $m = 1, 2, \cdots, M-1$.

Since Q_m is larger when m is smaller, the STB queue should be in as high a layer as possible from the bandwidth consumption point of view. Therefore, the layer interval $l_{m+1} - l_m$ for small m should be small. It is indicated by (11) that the layer interval is restricted by ϵ_m. Thus ϵ_m should be as small as possible for small m. So we introduce a parameter κ ($\kappa = 0, 1, \cdots, M-1$), and set $\epsilon_m = 0$ for $1 \leq m \leq \kappa$. Since a large ϵ_m increases the FRP interval, we should make ϵ_m as large as possible for large m. Therefore, we introduce a parameter α ($1.0 < \alpha$) and set ϵ_m for $\kappa+2 \leq m \leq M-1$ as

$$\epsilon_m = \epsilon_{\kappa+1}\alpha^{m-\kappa-1}, \quad (13)$$

where $\epsilon_{\kappa+1}$ is set as the maximum value of the right side of (12):

$$\epsilon_{\kappa+1} = \omega(Y_{max} - Y_{min}) - 1.$$

Since ϵ_m ($\kappa+2 \leq m \leq M-1$) is greater than $\epsilon_{\kappa+1}$, ϵ_m for arbitrary m satisfies (12).

4.2 Setting Q_1, Q_{M-2}, l_1, and l_{M-1}

Since Q_1 and Q_{M-2} can be set freely in the region of $Y_{min} < Q_{M-2} < Q_1 < Y_{max}$, let us introduce a parameter β ($0 < \beta < 0.5$) and set them when $M \geq 4$ as

$$\begin{cases} Q_1 & = (1-\beta)Y_{max} + \beta Y_{min}, \\ Q_{M-2} & = (1-\beta)Y_{min} + \beta Y_{max}, \end{cases} \quad (14)$$

and when $M = 3$ as

$$Q_1 = Q_{M-2} = (1-2\beta)Y_{min} + 2\beta Y_{max}. \quad (15)$$

If FRP starts in $x_n^+ = l_1 - \epsilon_1$ and $r_{n+\omega} = FQ_1$, the left inequality of (4) becomes

$$(\omega+1)FY_{max} - F(l_1-\epsilon_1) - \sum_{i=n}^{n+\omega-1} r_i \le FQ_1.$$

$$l_1 \ge max\left\{(\omega+1)Y_{max} - \sum_{i=n}^{n+\omega-1}\frac{r_i}{F} - Q_1 + \epsilon_1\right\}$$

is obtained. To minimize the required memory size at STB, l_1 is set as

$$l_1 = (\omega+1)\left(Y_{max} - Q_1\right) + \epsilon_1. \quad (16)$$

We can derive l_{M-1} from Q_{M-2} in the same way:

$$l_{M-1} = B + \omega Y_{min} - (\omega+1)Q_{M-2} - \epsilon_{M-1}. \quad (17)$$

4.3 Consideration of l_m

We have only to consider (11) for determining l_m ($m = 2, 3, \cdots, M-2$). Let μ denote the length of layer one ($\mu \equiv l_2 - l_1$). As mentioned before, μ should be small. From (11), we have

$$\mu = max\Big\{\epsilon_1 + (\omega+1)(Y_{max} - Y_{min}), \epsilon_2 + (\omega+1)(Y_{max} - Q_2)\Big\}.$$

To make the layer length $l_{m+1} - l_m$ larger as m increases, let us set l_m for $3 \le m \le M-2$ as

$$l_m = l_{m-1} + \mu \cdot \delta^{m-2} \quad (18)$$

where δ is a parameter derived from the buffer size B.

There is a minimum required value for δ to satisfy (11) for arbitrary m. Let $\delta_{req}^{(m^*)}$ denote the minimum δ satisfying (11) for $m = m^*$. From (11) and (18), $\delta_{req}^{(m^*)}$ is derived as

$$\delta_{req}^{(m^*)} = \left\lceil \frac{1}{\mu} max\Big\{\epsilon_m + (\omega+1)(Q_{m-1} - Y_{min}), \epsilon_{m+1} + (\omega+1)(Y_{max} - Q_{m+1})\Big\}\right\rceil^{\frac{1}{m-1}} \quad (19)$$

Moreover let us define δ_{req} as

$$\delta_{req} \equiv \max_{2 \le m^* \le M-2} \delta_{req}^{(m^*)}.$$

The parameter δ_{req} represents the minimum required value for parameter δ. The minimum required STB buffer size, B_{min}, is derived from (16) and (17) as follows:

$$B_{min} = \mu \cdot \frac{\delta_{req}^{M-2} - 1}{\delta_{req} - 1} + (\omega+1)(Y_{max} - Q_1 + Q_{M-2}) - \omega Y_{min} + \epsilon_1 + \epsilon_{M-1}. \quad (20)$$

Next, let us compute the parameter δ from a given B ($B \ge B_{min}$). From (16), (17), and (18), we can obtain the following equation for δ:

$$\delta^{M-2} - \Gamma\delta + \Gamma - 1 = 0 \quad (21)$$

$$\Gamma \equiv \mu^{-1}\Big\{B + \omega Y_{min} - (\omega+1)(Y_{max} - Q_1 + Q_{M-2}) - \epsilon_1 - \epsilon_{M-1}\Big\}.$$

4.4 Setting Q_m

We can use any value for Q_m ($2 \le m \le M-3$) so long as (10) is satisfied. For example, it is shown that the number of bits per frame obeys the exponential distribution in MPEG[12]. Let $H(y)$ denote the probability distribution function of the number of bits per frame, i.e.

$$H(y) = 1 - e^{-\frac{y}{Y_{av}}},$$

where Y_{av} is the average frame size. Now let us determine $H(Q_{M-2}), H(Q_{M-3}), \cdots, H(Q_1)$ in the same interval:

$$H(Q_m) = H(Q_1) - (m-1)\left(e^{-\frac{Q_{M-2}}{Y_{av}}} - e^{-\frac{Q_1}{Y_{av}}}\right)(M-3)^{-1}.$$

Q_m is calculated as

$$Q_m = -Y_{av} log\left\{e^{-\frac{Q_1}{Y_{av}}} + \frac{m-1}{M-3}\left(e^{-\frac{Q_{M-2}}{Y_{av}}} - e^{-\frac{Q_1}{Y_{av}}}\right)\right\}. \quad (22)$$

Q_m can be similarly calculated for other frame size distribution.

4.5 Initial situation

It is natural to make r_I equal to FQ_m ($1 \le m \le M-2$). In this case, x_d^+ is in the region:

$$Q_m d - Y_{max} \le x_d^+ \le Q_m d - Y_{min}.$$

Since x_d^+ should satisfy $l_m \le x_d^+ < l_{m+1}$, we obtain

$$Q_m^{-1}(l_m + Y_{max}) \le d < Q_m^{-1}(l_{m+1} + Y_{min})$$

when $r_I = FQ_m$ ($1 \le m \le M-2$). Since d should be as small as possible, let us set d as

$$d = \lceil Q_m^{-1}(l_m + Y_{max}) \rceil. \quad (23)$$

From (11), (16), and (17),

$$(d+\omega)r_I \ge (\omega+1)FY_{max} + F\Big\{\sum_{i=1}^{m}\epsilon_i + (Y_{max} - Q_m) + (\omega+1)\sum_{j=1}^{m-1}(Y_{max} - Q_j)\Big\},$$

$$(d+\omega)r_I < FB + \omega FY_{min} - F\Big\{\sum_{i=m+1}^{M-2}\epsilon_i + (Q_m - Y_{min}) + (\omega+1)\sum_{j=m+1}^{M-2}(Q_j - Y_{min})\Big\}.$$

So (9) is satisfied. Among $1 \leq m \leq M-2$, m is selected for r_I as small as possible at each call setup considering the available link capacity.

4.6 Protocol description

If the queue length is in layer $m = 0$ or $M-1$, AM1 can be used. Now, we describe the advanced protocol. This protocol is named DAQS-Multi-Layer Protocol (DAQS-MLP).
[DAQS-MLP]

Line 1 Transmission starts at $t = 0$ with a fixed rate of r_I. Here, r_I satisfies $r_I = FQ_{m^*}$ $(1 \leq m^* \leq M-2)$ and takes the maximum available value.

Line 2 Playback starts at $t = d (d = \lceil Q_{m^*}^{-1}(l_{m^*} + Y_{max}) \rceil)$. The layer indicator m is initialized as $m = m^*$.

Line 3 At the instance when one frame is removed from the buffer at $t = n$ $(d \leq n \leq V - \omega)$, the Algorithm Module 2 (AM2) shown in Fig. 5 is executed ($l_0 \equiv 0$, $l_M \equiv B$).

```
begin
  Set k=0, r_{n+w} = r_{n+w-1}
  if x_n < l_m - ε_m then  m=m-1, k=1
  else if x_n > l_{m+1} + ε_{m+1} then  m=m+1, k=1
  if (m=0) or (m=M-1) then
  begin
    if x_n < (w+1)Y_max - g_n(r) then
    r_{n+w} = (w+1)FY_max - f_n(x+, r),  Start FRP
    else if x_n > B+wY_min - g_n(r) then
        r_{n+w} = FB+wFY_max - f_n(x+, r),  Start FRP
  end
  else if k=1 then  r_{n+w} = FQ_m,  Start FRP
end.
```

Figure 5: Algorithm module 2

Some interactive operations change the video frame rate. We extend DAQS-SP and DAQS-MLP to interactive operations in [13]. The modified DAQS-SP is called DAQS-SPI (DAQS-SP with Interactive operation), and the modified DAQS-MLP is called DAQS-MLPI (DAQS-MLP with Interactive operation). In this paper, the description of the extension is omitted because of space limitation.

5 Performance Evaluation

In this section, we evaluate the performance of the proposed methods. Since no analytical models can adequately represent video traffic, the performance is evaluated by a computer simulator. The distance between video server and STB is assumed to be $50km$ ($\omega = \omega_{max} = \omega_{min} = 1$). Just one STB[†] and the video server is considered, and the simulation is performed for the whole movie *Star Wars*[14]. The evaluated value is averaged among 2500 trials.

Twelve successive frames form one segment. The frame rate, F, is 24 fps, the total number of frames, V, is 174128, the maximum frame size, Y_{max}, is 1.85267×10^5 $bits$, the minimum frame size, Y_{min}, is 476 $bits$, and the average frame size, Y_{av}, is 1.55983×10^4 $bits$.

For the sake of simplicity, only two interactive operations are considered: the slow forward and the fast forward with segment skipping. A fixed slow playback frame rate of 3 fps is used. In the fast playback mode, seven segments are skipped after playingback one segment. Thus this fast playback corresponds to 24×8 fps.

The user alternates between normal playback and interactive operation. The duration of normal playback obeys the exponential distribution with mean value $900s$. The duration of interactive operations also obeys the exponential distribution with mean value $10s$. At each interactive operation, the fast playback and the slow playback are selected with equal probability.

5.1 Comparison with other methods

First, the CoV of the transmission rate of DAQS-SPI and DAQS-MLPI are compared with other dynamic bandwidth allocation methods. In particular, one of them preserves the frame period. We call this method BRFP (Bandwidth Renegotiation per Frame Period). In BRFP, the bandwidth is negotiated in every frame period. Besides, the optimal smoothing method[8] (OPT) is also evaluated. It is difficult to apply OPT to the IVOD system because of its complex pre-calculation; however, we can use its performance as an ideal case for the CoV of the transmission rate. The average pre-loading delay in the optimal smoothing is set to the same value in DAQS ($500ms$). The bandwidth allocation table in OPT is re-calculated when the interactive operation starts or ends. The values of the four parameters in DAQS-MLPI are summarized in Table 1 (the influence of these four parameters on performance is evaluated in [13]). M is the number of layers. The layer boundary margin ϵ_m is set to be zero for $1 \leq m \leq \kappa$. The parameter α determines the layer boundary margin ϵ_m. The parameter β determines Q_1 and Q_{M-1} through (14) and (15). Here, we assume four STB buffer sizes, 4 M, 8 M, 12 M, and 16 $Mbytes$.

Table 1: Values for DAQS-MLPI parameters

B ($Mbytes$)	4	8	12	16
M	11	14	13	14
κ	1	2	2	2
α	1.400	1.275	1.425	1.450
β	0.064	0.070	0.070	0.070

In Table 2, the CoV of the transmission rate is summarized. Since the allocated bandwidth takes two extreme values in DAQS-SPI, the CoV of the transmission rate is

[†]Since we assume FRP request will not be rejected, we do not need to simulate multiple STBs.

Table 2: Comparison of the CoV of the transmission rate

B ($Mbytes$)	4	8	12	16
BRFP	1.1652	1.1652	1.1652	1.1652
DAQS-SPI	1.0721	1.1889	1.2849	1.3605
DAQS-MLPI	0.2471	0.2242	0.2126	0.2094
OPT	0.1575	0.1385	0.1336	0.1252

very large compared with DAQS-MLPI and the optimal value. It should be noted that DAQS-MLPI shows a good performance close to the optimal. For example, the CoV in DAQS-MLPI is 0.2471 and the optimal value is 0.1575 when $B = 4.0$ $Mbytes$.

Table 3: Comparison of bandwidth utilization

B ($Mbytes$)	4	8	12	16
PRA	0.0842	0.0842	0.0842	0.0842
MBA	0.7454	0.8161	0.8981	1.0
DAQS-SPI	0.9680	0.9459	0.9184	0.8949
DAQS-MLPI	0.9706	0.9439	0.9317	0.9189
OPT	0.9785	0.9598	0.9449	0.9357
BRFP	1.0	1.0	1.0	1.0

In Table 3, the bandwidth utilization in DAQS-SPI and DAQS-MLPI are compared with other bandwidth allocation methods in VOD. Since the required STB memory size in CRTT is much larger than assumed memory size, the minimum bandwidth allocation method[6] called MBA (Minimum Bandwidth Allocation) and OPT are evaluated. We also show the case of BRFP and the Peak Rate Allocation (PRA) method. Generally speaking, increase in the STB memory makes the allocated bandwidth larger. So the bandwidth utilization in DAQS and OPT degrades as the memory size increases. Although MBA may show good performance when the STB memory size is large, huge pre-loading may be required (it depends on the video source and the memory size. In the case of $16Mbytes$ STB memory, the pre-loading delay is $37s$). On the other hand, DAQS-SPI and DAQS-MLPI require short pre-loading delay ($500ms$).

From Tables 2 and 3, it is concluded that DAQS-MLPI has excellent characteristics compared with other bandwidth allocation methods in VOD.

6 Conclusion

A new bandwidth allocation method for the IVOD system with constant quality is proposed. This method, called DAQS, dynamically renegotiates the allocated bandwidth based on the STB queue length to avoid both buffer underflow and overflow. No complex pre-calculation is necessary, so this method can be applied to IVOD. Moreover, the multi-layer concept is introduced to decrease the CoV of the transmission rate.

Through a simulation model, it is shown that the CoV of the transmission rate in DAQS-MLPI is close to the optimal and the bandwidth utilization is close to unity. So we conclude that DAQS-MLPI is an excellent bandwidth allocation method suited for IVOD. It is noted that this method can be used for any stored video service with interactivity.

References

[1] Y. Chang et al. "An Open-Systems Approach to Video on Demand". *IEEE Communication Magazine*, pages 68–80, 5 1994.

[2] D. Deloddere, W. Verbiest, and H. Verhille. "Interactive Video On Demand". *IEEE Communication Magazine*, pages 82–88, 5 1994.

[3] D. Reininger and D. Raychaudhuri. "Bit-Rate Characteristics of a VBR MPEG Video Encoder for ATM Networks". In *IEEE ICC'93*, pages 517–521, 1993.

[4] C. Huang, M. Devetsikiotis, I. Lambadaris, and A.R. Kaye. "Modeling and Simulation of Self-Similar Variable Bit Rate Compressed Video: A Unified Approach". In *ACM SIGCOMM'95*, pages 114–125, 1995.

[5] J.M. McManus and K.W. Ross. "Video-on-Demand Over ATM: Constant-Rate Transmission and Transport". *IEEE Journal on Selected Areas in Communications*, 14(6):1087–1098, 8 1996.

[6] J. Lauderdale and D.H.K. Tsang. "Using the Minimum Reservation Rate for Transmission of Pre-Encoded MPEG VBR Video Using CBR Service". *IEICE Trans. Communications*, pages 1023–1029, 8 1996.

[7] M. Grossglauser, S. Keshav, and D. Tse. "RCBR: A Simple and Efficient Service for Multiple Time-Scale Traffic". In *ACM SIGCOMM'95*, pages 114–125, 1995.

[8] J.D. Salehi, Z. Zhang, J.F. Kurose, and D. Towsley. "Supporting Stored Video: Reducing Rate Variability and End-to-End Resource Requirements through Optimal Smoothing". In *ACM SIGMETRICS'96*, 5 1996.

[9] H. Zhang and E. Knightly. "A New Approach to Support Delay-Sensitive VBR Video in Packet-Switched Networks". In *5th Workshop on Networking and Operating System Support for Digital Audio and Video*, pages 275–286, 4 1995.

[10] S. El-Henaoui, R. Coelho, and S. Tohme. "A Bandwidth Allocation Protocol for MPEG VBR Traffic in ATM Networks". In *IEEE INFORCOM'96*, pages 1100–1107, 3 1996.

[11] H. Shimonishi, T. Takine, M. Murata, and H. Miyahara. "Performance Analysis of Fast Reservation Protocol with Generalized Bandwidth Reservation Method". In *IEEE INFORCOM'96*, pages 758–767, 3 1996.

[12] C.S. Freedman and D.J. DeWitt. "The SPIFFI Scalable Video-on-demand System". In *ACM SIGMOD'95*, pages 352–363, 1995.

[13] N. Kamiyama and V.O.K. Li. "Dynamic Bandwidth Allocation based on Queue Length of STB in IVOD System". *submitted to IEEE/ACM Trans. on Networking.*

[14] M.W. Garrett and W. Willinger. "Analysis, Modeling and Generation of Self-Similar VBR Video Traffic". In *ACM SIGCOMM'94*, pages 269–280, 1994.