論文 98-23-3-21

Bandwidth Smoothing for VBR Video Streams in Video-On-Demand Service

Jae-Hong Lee*, Sun-Sook Lee*, Tae-Kyun Kim* *Regular Members*

ABSTRACT

Variable Bit Rate (VBR) video data show highly bursty traffic characteristics depending on their frame types. In order to transmit video streams in the ATM-based distributive Video-On-Demand service, a Bandwidth Smoothing scheme prefetching sufficient data from the server can be provided for the efficient utilization of the network before the playback of the video at a client.

We propose a new Bandwidth Smoothing scheme called the *Long Stretching in the Middle (LSM) bandwidth allocation algorithm* for the transmission scheduling of the stored VBR video streams in a large storage device of the server. And we present the performance metrics of the Bandwidth Smoothing Algorithm and analyze them according to the service types for the bandwidth reservation, and compare the performance metrics of the proposed algorithm with other algorithms based on bit traces of the MPEG-1 frames. The proposed LSM is a simple algorithm with O(N) complexity, which shows low average bit rate, small bandwith changes, small bandwidth increases and low bandwidth change frequency.

On the other hand, finding moderate buffer size and startup latency considering jitter and delay in the network and reducing demands for high bandwidth in the first segment are discussed.

Ⅰ. Introduction

As the result of the rapid development of the ATM based B-ISDN and data compression techniques such as JPEG and MPEG, the multimedia services with an enormous data have begun to permeate the market-place. Recently, a number of studies have focused on the Video-On-Demand (VOD) service in particular [1, 2].

VOD system is composed of server, client, and network connecting them. VOD server maintains the compressed video and audio data in a large storage device and delivers the data to clients through the ATM network or cable network. VOD client is a workstation or Set-Top-Box with MPEG decoder, and has a fixed-size buffer to receive data from VOD server.

Generally, VOD service can be classified into two categories: the *interactive* service, which allows a client's interactions such as rewind, fast forward and stop of the VCR functions during the playback, and the *distributive* service, which does not allow a client's requests for interactions until the full completion of a video playback.

Since the amount of video data is immense as compared to the audio data, transmitting the raw data over the network is inadequate. Consequently, for the efficiency of storage, transmission, and main-

*충남대학교 컴퓨터공학과
論文番號:97438-1128
接受日字:1997年 11月 28日

tenance of the constant-quality during the playback, video data need to be compressed using the MPEG scheme with VBR mode [3]. However, traffic characteristics of the compressed VBR MPEG video data show high burstiness due to the differences in bit rates between I frames with the spatial coding and P and B frames with the predictive coding. Thus, transferring such a bursty traffic over the network requires large bandwidth having high transmission costs and causes the traffic control difficult [4, 5]. In a stored video, unlike in a real-time video, since the statistical characteristics of a full stream is known before the transmission and prefetching sufficient data from the server before the playback is possible, using the minimum bandwidth in bandwidth smoothing for transmitting video streams is efficient for the utilization of the network [6, 13].

The bandwidth allocation scheme for a stored video can be classified into two categories as shown in Fig. 1 : the *static bandwidth allocation* for the fixed bandwidth of a full video stream [7, 8], and the *dynamic bandwidth allocation* considering a client-side buffer occupancy [9, 10, 11, 12, 15]. The dynamic bandwidth allocation can be reclassified into the *identical segment allocation* and *variable segment allocation*. In the identical segment allocation, the lowest fixed bandwidth is allocated to each segment without

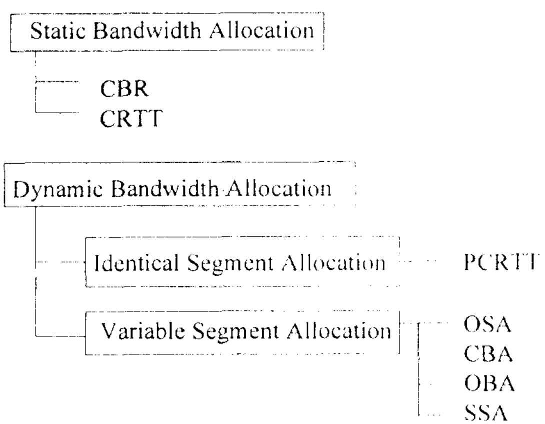


Fig. 1 Classification for the bandwidth allocation

generating buffer underflow [9]. And in the variable segment allocation, segments with variable lengths are created by finding optimal bandwidth allocation schedule to a limited extent that the buffer occupancy at a client does not generate buffer overflow or underflow [10, 11, 12, 15].

In the static bandwidth allocation, it is important to find the minimum transmission rate required for the transmission of a video with initial startup latency and to search for the minimum buffer size of a client-side at that rate.

However, during the playback of a full video stream, since a fixed bandwidth is allocated, it is advantageous that the server provides service without a renegotiation for the bandwidth if the requested bandwidth by Call Admission Control(CAC) is allocated.

In the dynamic bandwidth allocation, unlike in the static bandwidth allocation, the bandwidth negotiation step requesting CBR service based on a new bandwidth is required, and then the bandwidth currently allocated is released. Renegotiated CBR (RCBR) is an example of the bandwidth negotiation technique [17].

In this paper the *Long Stretching in the Middle* (LSM) bandwidth allocation algorithm is suggested as a bandwidth smoothing scheme for the transmission scheduling for the VBR video data in the distributive VOD service. In LSM, before VBR video traffics are transmitted to the User Network Interface (UNI), a smoothing is conducted at the server by using the frame sizes of the video stream.

LSM finds the long-lasting segments that last for a long time without buffer overflow or underflow in order to lower average bit rate, bandwidth change number and to maximize the length of a segment. LSM reduces bandwidth change number and bandwidth increase number, and average bit rate with a simple algorithm by eliminating the overhead produced in the previous schemes using a linear search looking for the starting point when the bandwidth changes.

This paper is organized as follows. In section Ⅱ, we discuss the bandwidth smoothing and related works. In section Ⅲ, we propose a new bandwidth smoothing algorithm called LSM. In section Ⅳ, we present performance metrics for the evaluation of performance of the bandwidth smoothing algorithm and describe the importance of each performance metrics according to the service types. In section Ⅴ, we compare and analyze proposed LSM algorithm and previous algorithms based on the performance metrics employing the MPEG-1 video data, and conclude in section Ⅵ.

Ⅱ. Bandwidth Smoothing and Related Works

1. Bandwidth Smoothing

In VOD server, transmission scheduling of a video stream is written by the bandwidth smoothing technique before the transmission. A bandwidth for a full video stream is fixedly allocated by the static bandwidth allocation [7, 8] or dynamically allocated by the dynamic bandwidth allocation [9, 10, 11, 12, 15]. Since the static bandwidth allocation requires a huge buffer and brings intolerable initial startup latency to lower the bandwidth requirements, in this paper as the transmission schedule dynamic bandwidth allocation is applied. In dynamic bandwidth allocation, the constant bandwidth is applied for a specific segment composed of multiple frames.

VOD server finds the starting frame number, length and bandwidth of a segment by the bandwidth smoothing scheme, and maintains the transmission schedule based on the above information on a storage device with a video stream.

A video stream V, composed of K consecutive segments Si is expressed in equation 1.

$$V=\{S_1, S_2, \cdots, S_i, \cdots, S_K\} \tag{1}$$

The transmission schedule of segment S_i is shown in equation 2.

$$S_i=\{F_i, L_i, R_i\} \tag{2}$$

In equation 2, F_i, L_i, R_i represent the starting frame number, segment length, and allocated bandwidth, respectively.

VOD server transmits the video data to a client using the transmission schedule as represented in equation 1, with the bandwidth R_i from the starting frame F_i of the segment Si to the F_i+L_i.

After VOD server completes the transmission of the frames of segments S_i, the server tries to negotiate with the network to reserve a new required bandwidth, R_{i+1}, before transmitting the frames of the next segment, S_{i+1}. And then, the reserved bandwidth for the segment S_i is released to the network, and a new bandwidth R_{i+1} is required in the network. And the bandwidth negotiation is conducted to find out if the network provides the requested bandwidth. For this, Renegotiated Constant Bit Rate (RCBR) service can be employed.

The main purpose of the bandwidth smoothing is to find the optimal bandwidth allocation schedule in order to transmit the video stream within given constraints: client-side buffer size and initial startup latency without buffer overflow or underflow. Another purpose of bandwidth smoothing is to optimize the performance metrics as discussed in section Ⅳ.

A VBR video stream V is composed of N frames whose ith frame size is f_i. In order to guarantee continous playback at a client, the server must always transmit the video data to prevent client's buffer underflow. That is, the server should transmit more than F_{lower} data to a client.

$$F_{lower}(n)=\sum_{i=1}^{n} f_i \tag{3}$$

Meanwhile, if the client has a fixed-length buffer the server should not transmit more than B, which is the amount of data that the buffer can accept. In other words, the client should not receive more than F_{upper}, at the frame position n.

$$F_{upper}(n) = F_{lower}(n) + B \tag{4}$$

On the other hand, the client-side buffer size has to be greater than the maximum frame size of a video stream.

$$B \geq \max_{1 \leq i \leq N} f_i \tag{5}$$

The bandwidth allocation schedule of a server searches for the optimal slopes of curves located between F_{lower} and F_{upper}, that satisfies specific performance metrics on which we will discuss in section Ⅳ. That is, if the nth frame belongs to the kth segment and R_k is the allocated bandwidth to the segment, F_k frame is the starting frame number of the segment, the accumulated data that the client have received from the server through nth frame should satisfy equation 6.

$$C(k-1) = \sum_{i=1}^{k-1} (R_i \times L_i)$$
$$F_{lower}(n) \leq C(k-1) + R_k \times (n - F_k) \leq F_{upper}(n) \tag{6}$$

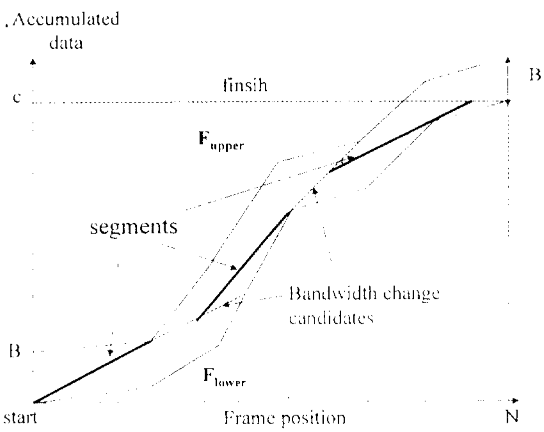


Fig. 2 shows an example of bandwidth allocation schedule of server.

2. Related Works

Piecewise Constant Rate Transmission and Transport (PCRTT)[9] suggests a transmission scheduling that requires the minimum buffer for multiple segments with the same interval. However, PCRTT intensifies the bandwidth changes between segments dividing a stream into the random fixed numbers of segments without regard for the characteristics of the video stream. The shortcomings of PCRTT is that bandwidth changes have to be increased for the reduction of requests for a buffer.

In the variable segment allocation, there are number of schemes such as Optimal Smoothing Algorithm (OSA) by Salehi [10], Critical bandwidth Allocation (CBA) and Optimal Bandwidth Allocation (OBA) by Feng [11, 12], and Sliding Stretching Algorithm (SSA) by Zhang [14]. They can be classified by the selection methods for bandwidth change time, which is one of the bandwidth change candidates represented by the dotted line in Fig. 2 to look for the initial starting point of the segment S_{i+1} to maximize the length of segment S_i.

In OSA, data are transmitted in CBR for a long possible time without buffer overflow or underflow. And to make the bandwidth variability small the bandwidth has to be changed as early as possible. Accordingly, bandwidth changes only in the case that the client-side buffer is full or empty. That is, in Fig. 2, among the bandwidth change candidates, the leftmost frame position is selected as a change point. And the maximum complexity of OSA shows O(N2) [10]. This is due to the potential backtracking in the algorithm.

In CBA, when a new bandwidth is allocated, a linear search is conducted to minimize the number of increases for the bandwidth. In OBA, linear search is used to minimize the bandwidth changes even in the case that currently allocated bandwidth is decreased. In CBA and OBA, the initial startup latency is not considered, and the complexity of these schemes is $O(N^2 \log N)$, where the log N term arises from performing a binary search along the bandwidth change candidates of each segment [13].

In SSA, bandwidth R_1 is found to maximize the length of the first segment S_1 by conducting a linear search on all the points of a straight line connected by F_{lower} and F_{upper} at the start time as in Fig. 2. Accordingly, SSA requires the initial buffer buildup at the start frame position. The identical methods as in OBA are used for the initial starting point of each segment after S_2. When C is the cumulative data of a video stream, that is $C=\sum_{i=1}^{N} f_i$, the complexity of SSA is O(NxC). And a proof of the complexity O(NxC) is shown in [16].

Ⅲ. Long Stretching in the Middle (LSM) Bandwidth Allocation

1. LSM Algorithm

The mathematical model of LSM we propose in this paper is shown in Fig. 3. LSM is an algorithm to find $R(n)$ at any frame position n for the optimal bandwidth allocation in the region encompassed with two curves, where $c=P(n)$, $c=A(n)$ in Fig. 3.

$R(n)$ is a curve composed of piece-wise lines, where the slope of each line becomes the allocated bandwidth. And a fixed bandwidth is allocated for a segment. In Fig. 3, t_s is the initial startup latency. Thus, the server could lower the bandwidth by transmitting video data t_s seconds before a client starts to playback video stream.

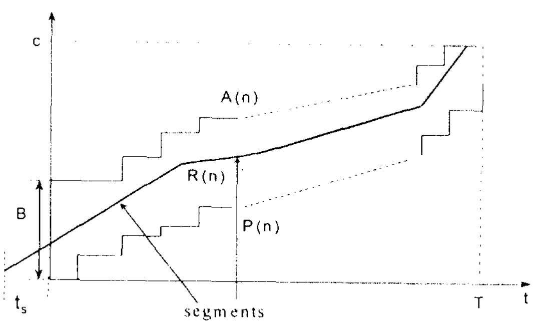


Fig. 3 LSM mathematical model

If N is the total number of frames of a video stream V, and Δ is the difference in playback time between two adjacent frames, then T is the total payback time of V which equals to $N \times \Delta$. Meanwhile, the playback time t is denoted by $t=n\times\Delta$ at the frame position n.

C is the cumulative data of a video stream V and expressed in equation 7.

$$C=\sum_{i=1}^{N} f_i = P(N) \tag{7}$$

The $F_{lower}(n)$ and $F_{upper}(n)$ at the frame position n in Fig. 2 is depicted as $P(n)$ and $A(n)$ in Fig. 3, respectively. The cumulative data playbacked by the client over [0, n] is $P(n)$, which is the total amount of frame sizes through nth frame.

$$P(n)=\sum_{i=1}^{n} f_i,\ P(0)=0 \tag{8}$$

If the buffer size is B, the cumulative data $A(n)$, which the client can receive over [0, n] without buffer overflow, is shown in equation 9. In order for a client to playback nth frame the server should prefetch and transmit the data before the playback. Thus P(n-1) is used in equation 9.

$$\begin{aligned} &A(0)=B \\ &A(n)=\min\{B+P(n-1),\ C\},\ 1\le n\le N \end{aligned} \tag{9}$$

Since $R(n)$ is the cumulative data transmitted by the server over [0, n] to a client according to the transmission schedule, the slope becomes the bandwidth of a segment to which the nth frame belongs.

The followings are the definitions of a straight line between two points, the slope of a straight line, and P_i and A_i.

Definition 1 : When $x2\ge x1$ and $y2\ge y1$, a straight line between point $p1=(x1, y1)$ and point $p2=(x2, y2)$

is denoted as [p1 → p2], and the slope of which is denoted as SL([p1 → p2]).

Definition 2: Point (i, P(i)) and point (i, A(i)) are denoted as P_i, A_i, respectively.

The $max\_upper(n_0, n)$ is the maximum bandwidth not generating buffer overflow from the starting frame n_0 to an arbitrary frame n. And the $min\_lower(n_0, n)$ is the minimum bandwidth that does not generate buffer underflow from the starting frame n0 to an arbitrary frame n.

$$P_i = (i,\ P(i)),\ A_i = (i,\ A(i)),\ R_0 = (n_0,\ R(n_0))$$

$$max\_upper(n_0, n) = \min_{n_0 \le i \le n} SL([R_0 \to A_i]) \qquad (10)$$

$$min\_lower(n_0, n) = \min_{n_0 \le i \le n} SL([R_0 \to P_i])$$

$SL_{half}(n_0, n)$ indicates the slope of the mean value of $A(n)$ and $P(n)$ at an arbitrary frame n from the starting frame n_0.

$$SL_{half}(n_0, n) = SL[R_0 \to p] \qquad (11)$$

$$where\ p = (n, \frac{A(n) + P(n)}{2})$$

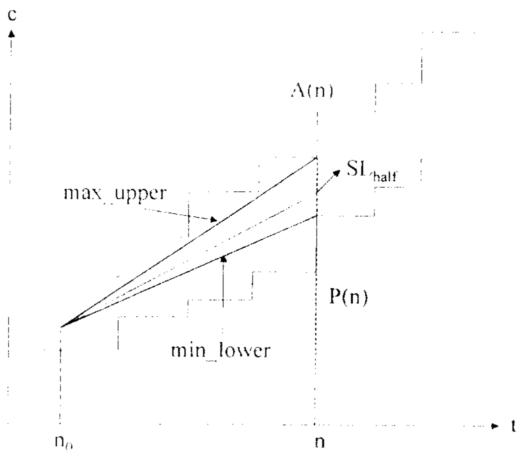


Fig. 4 max_upper, min_lower, SL_{half} of LSM

In Fig. 4, the maximum value, which satisfies equation 12 from the starting frame n_0 of the segment S_i to an arbitrary frame n, becomes the end point of the segment S_i, and this point becomes the starting point of segment S_{i+1}.

In LSM, in order to maximize the length of a segment and reduce bandwidth variability, the frame position which satisfies the condition in equation 12 is selected as the end point of the segment, and the bandwidth R_i is allocated for the segment S_i as in equation 13 from frame n_0 to frame n. In equation 13, R_{play} represents the playback rate, whose unit is the number of frames per second.

$$min\_lower(n_0, n) \le SL_{half}(n_0, n) \le max\_upper(n_0, n) \qquad (12)$$

$$R_i = SL_{half}(n_0, n) \times R_{play} \qquad (13)$$

LSM algorithm is shown in Fig. 5. LSM performs only a linear search on all frames to find segments. Thus if the total number of frames of a video stream V is N, the complexity of LSM is O(N).

```
Input  : Initial point p0 = (x0, y0)
Output : End point p = (x, y) and Rate Ri (bits per frame)
         of a segment Si
/* Initialization */
n ← x0 + 1;
r_max ← ∞;
r_min ← 0;
while( n ≤ N ) {
  r1 ← max_upper(x0, n);
  r2 ← min_lower(x0, n);
  if ( r1 < r_max ) then r_max ← r1;
  if ( r2 > r_min ) then r_min ← r2;
  r_half ← SL_half(x0, n);
  if ( r_max ≥ r_half ) ∧ ( r_half ≥ r_min )
    then n ← n + 1;   /* Equation 12 */
    else {
          x ← n - 1;
          y ← (A(x) + P(x)) / 2
          r_half ← SL(x0, x);
          return(p, r_half);
    }
}
/* last segment */
x ← n;
y ← r_half × (n - x0) + y;
return(p, r_half);
```

Fig. 5 Pseudo code for LSM algorithm

As LSM finds $R(n)$ passing through the mid-point of two curves, $A(n)$ and $P(n)$, to maximize the lengh of a segment and to reduce bandwidth variability, the problem occurs when the bandwidth of the last segment S_K is allocated. In other words, it is a subject of controversy that the offset between the problem of reducing bandwidth variability and lowering average bit rate. In order to reduce bandwidth variability if we take SL_{half}, average bit rate might become higher; or if min_lower value is taken to lower the average bit rate, bandwidth variability might become greater. In LSM we take SL_{half} to reduce bandwidth variability.

2. Initial Startup Latency

In order to lower average bit rate and reduce bandwidth variability, video data should be smoothed and transmitted by the server t_s seconds before a client starts to playback a video stream. In Fig. 3, to lower the average rate and reduce bandwidth variability by a bandwidth smoothing scheme, the optimal initial startup latency t_s can be calculated as the following.

Step 1: In Fig. 6, the first segment S_1 is found by using LSM algorithm with all the points(1, i) on the straight line $[P_1 \rightarrow A_1]$, and the length $L_1(i)$ of the segment S_1 for each point is calculated.

Step 2: The slope R_1 is found at the frame position m with the maximum length L_{max} of S_1.

$$L_{max} = \max_{P(1) \le i \le A(1)} L_1(i)$$

$$R_1 = SL([P_1 \rightarrow P']) \tag{14}$$

$$where\ P' = \left(m, \frac{A(m)+P(m)}{2}\right)$$

Step 3: The point adjacent to t-axis for the straight line with slope R_1 is found as in Fig. 6. This point becomes the optimal initial startup latency t_s.

$$\frac{A(m)+P(m)}{2} = R_1 \times (t_s + m \times \Delta)$$

$$t_s = \frac{A(m)+P(m)}{2 \times R_1} - m \times \Delta \tag{15}$$

$$where\ \Delta = \frac{1}{R_{play}}$$

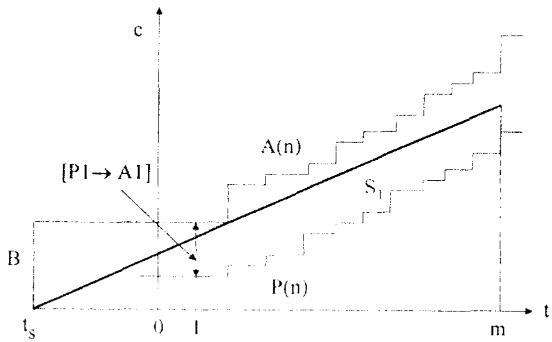


Fig. 6 The optimal initial startup latency

If t_s of the video stream V is found using equation 15, it is easy to obtain the optimal initial startup playback latency without searching for the optimal bandwidth smoothing schedule for various initial startup latencies.

3. Jitter Considerations

Due to the delay and jitter in the network, video data are not able to arrive on playback time according to the transmission schedule of the server. Accordingly, buffer underflow or overflow can be generated at a client.

In LSM, in order to absorb data, which the client can receive during the jitter(the difference between maximum delay d_{max} and minimum delay d_{min}), equation 5 is modified to equation 16. In equation 16, R_{max} is the maximum value among the bandwidths allocated to each segement. In order to absorb delay and jitter in the network, if the maximum delay d_{max} is considered the initial startup latency t_s is calculated as in equation 17. In order to absorb jitter for the limited buffer B equation 9 has to be modified to equation 18.

$$\delta = d_{\max} - d_{\min}$$
$$B' \geq B + \delta \times R_{\max} \tag{16}$$

$$t_s' = t_s + d_{\max} \tag{17}$$

$$A(n) = \min\{ B' + P(n-1), C \},\ 1 \leq n \leq N \tag{18}$$

Ⅳ. Performance Metrics of the Bandwidth Smoothing

1. Performance Metrics

In this section, the factors for the performance evaluation of the bandwidth smoothing are introduced.

1) Peak bit rate

Peak bit rate R_{max} is the maximum value among the bandwidths allocated to each segment. For the efficient utilization of the network, the peak bit rate has to be small. A calculation to find peak bit rate is shown in equation 19.

$$R_{\max} = \max_{1 \leq i \leq K} R_i \tag{19}$$

2) Average bit rate

Average bit rate R_{avg} is the mean value of the bandwidths allocated to each segment. Average bit rate can be calculated as in equation 20.

$$R_{avg} = \frac{1}{K} \sum_{i=1}^{K} R_i \tag{20}$$

3) Bandwidth changes

K is the Bandwidth changes, that is the total number of segments found by a bandwidth smoothing. As it influences the bandwidth renegotiation number, small number is preferable.

4) Bandwidth increases

In case that the bandwidth R_{i+1} of a segment S_{i+1} requires larger bandwidth than currently allocated bandwidth R_i, the requested bandwidth R_{i+1} can not be allocated at the negotiation step for a bandwidth reservation. Accordingly, the bandwidth increases also has to be small for the same bandwidth changes. Meanwhile, in case that the bandwidth decreases the allocated bandwidth is returned to the network and smaller bandwidth is requested. Accordingly, the probability for the failure of bandwidth reservation at the renegotiation step becomes smaller.

5) Bandwidth variability

The bandwidth variablity R_{std} shows the degree of smoothness for the transmission schedule. R_{std} has to be small to prevent interruption of the service or degradation of Quality of Service (QoS) in case the requested bandwidth is not available at the bandwidth renegotitation step. The bandwidth variability is the standard deviation of the allocated bandwidths as shown in equation 21.

$$R_{std} = \sqrt{\frac{1}{K} \sum_{i=1}^{K} (R_i - R_{avg})^2} \tag{21}$$

6) Bandwidth Change Frequency

Bandwidth Change Frequency indicates the degree of how often bandwidth changes during the playback of a full video stream, and it is inversely proportional to the average length of segments. As the average length of segments becomes longer, the time in which video data are transmitted without renegotiation of bandwidth becomes longer.

Bandwidth Change Frequency R_{freq} can be obtained by equation 22.

$$R_{freq} = \frac{R_{play}}{\frac{1}{K} \sum_{i=1}^{K} L_i} \tag{22}$$

2. The Importance of Performance Metrics according to Bandwidth Reservation Types

The following service types are considered for the bandwidth reservation for VOD service in the ATM network. (1) CBR type which uses a fixed bandwidth,

(2) RCBR type which uses the dynamic bandwidth allocation and requires the renegotiation step for the bandwidth reservation, (3) VBR type which uses statistical multiplexing, and (4) Available Bit Rate [18] type, which provides the limited QoS within available bandwidth.

CBR type allocates a fixed bandwidth during the playback of a full video stream applying the peak bit rate. If bandwidth variability is large, it is not efficient for the resource utilization because it produces a large waste of bandwidth. Therefore, the peak bit rate is not an important factor for any type except CBR type.

VBR type is suitable for the applications with bursty traffic pattern and is not efficient for non-realtime application like VOD services. In VBR type, statistical multiplexing gain is small for a smoothed traffic [19].

In ABR type, which provides the limited QoS within available bandwidth, even though there is no probability for failure of bandwidth renegotiation, user's requests for QoS are not fully satisfied because there are cases that the required bandwidth can not be provided. In case of a video stream which is encoded using layered encoding of MPEG-2, ABR type can be used. Thus, RCBR type is considered as the most powerful type for a resource reservation method to support VOD service within a given buffer size and tolerable initial startup latency. In RCBR type, in order to minimize the cost for the renegotiation, bandwidth changes have to be small. And also, to reduce connection cost of the network, transmission time should be short. On the other hand, to reduce the probability for the failure both bandwidth variability and bandwidth increases should be small at the bandwidth negotiation step.

V. Experiments and Evaluations

The results are compared and evaluated by using the bit traces of MPEG-1 coded "star wars" [20] and "lambs" [21] for the performance evaluation of the proposed bandwidth smoothing (LSM).

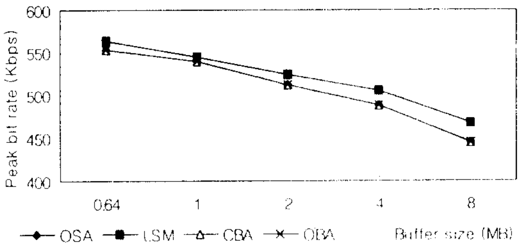
Table 1 shows the encoding characteristics for "star wars" and "lambs" used in our experiments.

Table 1. Encoding characteristics for "star wars" and "lambs"

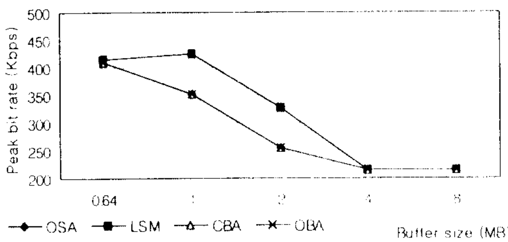
Video streams	star wars	lambs
Resoultion	508 x 408	384 x 288
Total frame length (frames)	174,136 (119 minutes)	40,000 (27 minutes)
GOP(Group Of Picture) size	12	12
R_{play}	24	25
Average bit rate (Mbps)	0.374	0.175
Peak bit rate (Mbps)	4.446	3.221

In this paper Optimal Smoothing Algorithm(OSA), Critical Bandwidth Allocation (CBA), and Optimal Bandwidth Allocation (OBA) are selected for comparison. OSA uses majorization theory to minimize the bandwidth variability[10]. CBA and OBA search for the bandwidth change point by conducting a linear search among the candidates for the bandwidth change to minimize the bandwidth change number. As CBA and OBA both do not reflect the initial startup playback latency ts, equation 4 is used for F_{upper} in [11, 12]. But, we let $t_s = 10$ seconds for performance comparison under the same conditions as in OSA and LSM. For this, we use equation 9 instead of equation 4 as F_{upper} curve. The fluctuations of t_s show no great influences in performance metrics of the smoothing algorithms. But $t_s = 0$ indicates that there is no initial startup playback latency for the bandwidth R_1 of the first segment S_1. So R_1 becomes very huge because the first frame of the video stream is I frame and its size is very large. In this test, under the initial startup latency about 10 seconds, performances are compared.

The following figures demonstrate the performance results of OSA, LSM, CBA, and OBA in different buffer sizes with $t_s = 10$ seconds.

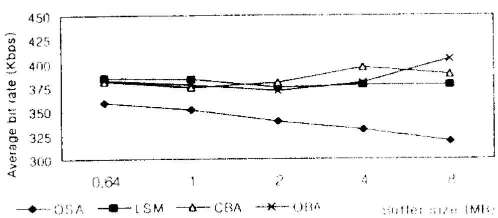


(a) star wars

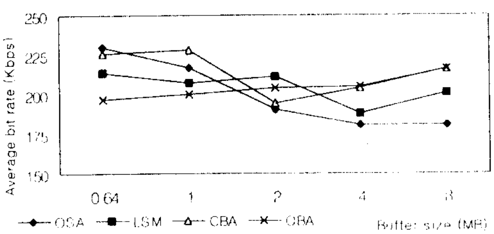


(b) lambs

Fig. 7 Peak bit rate for different buffer sizes

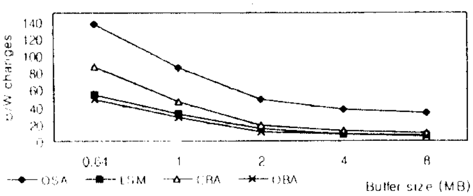


(a) star wars

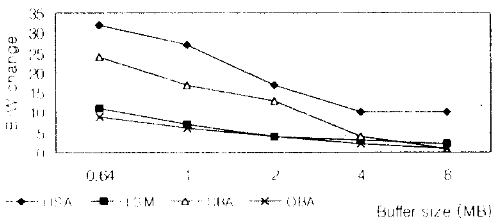


(b) lambs

Fig. 8 Average bit rate for different buffer sizes

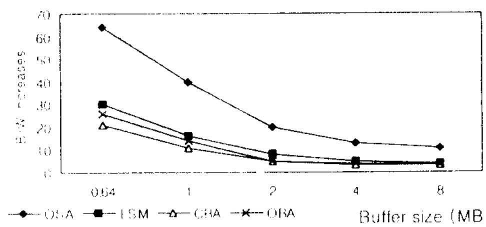


(a) star wars

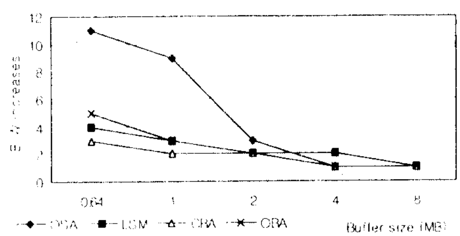


(b) lambs

Fig. 9 Bandwidth changes for different buffer sizes

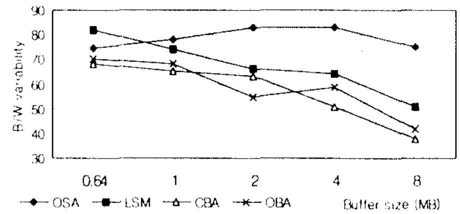


(a) star wars

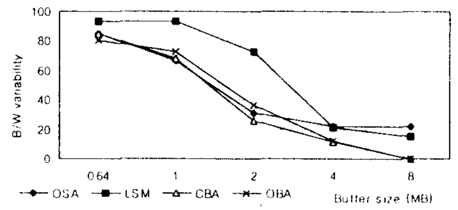


(b) lambs

Fig. 10 Bandwidth increases for different buffer sizes

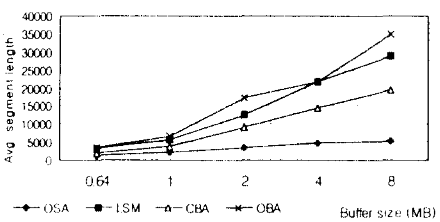


(a) star wars

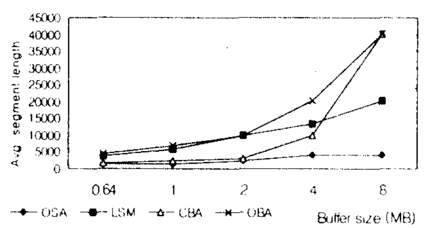


(b) lambs

Fig. 11 Bandwidth variability for different buffer sizes



(a) star wars



(b) lambs

Fig. 12 Average segment length for different buffer sizes

From the standpoint of peak bit rate, as shown in Fig. 7, OSA, CBA and OBA all have the same peak bit rate, and LSM has somewhat higher peak bit rate. While the peak bit rate is an important performance metrics for CBR type, average bit rate is important for RCBR type. As illustrated in Fig. 8, LSM has low average bit rate as compared to CBA and OBA. Considering average bit rate, OSA is the lowest.

In the aspect of bandwidth changes, as buffer size increases bandwidth changes of LSM, CBA, and OBA abruptly decreases, but as buffer size decreases bandwidth changes of OSA becomes very high. CBA demonstrates very low performance in "lambs", and LSM and OBA show similar performances. In "lambs" with thirty-minute run time, bandwidth changes of CBA and OBA is reduced to 1 for the buffer sizes over 4MB, and turns out to be CBR type, which is a static allocation.

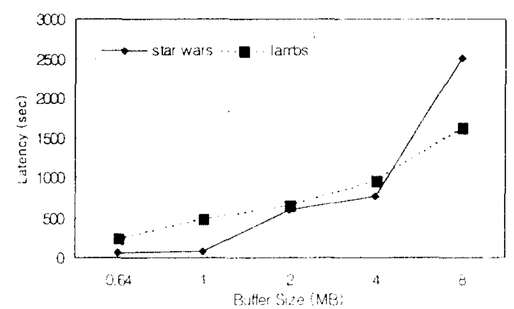
In regarding bandwidth increases, CBA, which minimizes bandwidth increases, demonstrates the most prominent performance. As the buffer size grows larger, LSM, CBA, and OBA show similiar performances. OSA has a large bandwidth changes and high bandwidth increases because OSA changes bandwidth at the buffer overflow or underflow.

In considering bandwidth variability, CBA displays the most excellent performance because it conducts a linear search for the candidates for the bandwidth changes to minimize this performance metrics. As the buffer size grows larger, OBA prefetches more data by increasing the average bit rate as shown in Fig. 8, and decreases the number of segments. As a result of the increase in average bit rate, as the buffer size grows larger, bandwidth variability of OBA becomes greater than CBA in "star wars". And LSM shows higher bandwidth variability than CBA and OBA even though LSM uses SL_{half}. In OSA, as the buffer size grows larger, bandwidth variability also grows larger. This is because the bandwidth variability grows larger as the buffer size grows since OSA changes bandwidth whenever the buffer overflow or underflow occurs.

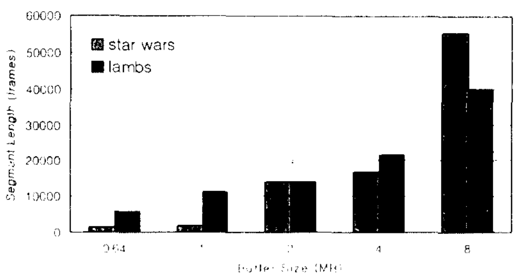
In the average length of segments, as the buffer size increases OBA shows the most outstanding performance and the proposed LSM demonstrates better performance than CBA and OSA. In "lambs", the curve shows lower value than CBA and OBA at 8MB buffer because one short segment appears at the end of a stream in LSM.

The result of performance evaluation shows that the proposed LSM is outstanding in average bit rate as compared to other algorithms. The bandwidth variability demonstrates higher performance in order of CBA, OBA, LSM, and OSA; and bandwidth changes and bandwidth increases show similar performances in LSM, CBA, and OBA, for the buffer sizes over 2MB. Since OSA changes bandwidth before the client-side buffer becomes overflow or underflow, many short segments appear. Moreover, OSA shows the lowest performance in performance metrics except in average bit rate as compared to other algorithms.

Fig. 13 shows the optimal initial startup latency for different buffer sizes in LSM. In "star wars", the optimal initial startup latency increases abruptly for buffer sizes over 2MB.



(a) Latency



(b) Segment Length (L_l)

Fig. 13 Optimal initial startup latency for different buffer sizes

When the number of frames is N, even though LSM has O(N) complexity it shows excellent or similar performances as compared to CBA or OBA, which has $O(N^2 \log N)$ complexity. For the future, there should be continuous studies for reducing bandwidth variability of LSM.

In the experiments, information on MPEG-1 frame sizes were used, and most of the data expect the "star wars" were about 30-minute run or JPEG coded, which were not sufficient for our experiments. Therefore, it is necessary to obtain full-length MPEG-2 data for further studies.

Ⅵ. Conclusions and Future Work

In this paper the *Long Stretching in the Middle* (LSM) bandwidth allocation algorithm, which is a dynamic bandwidth allocation scheme for VOD server to transmit VBR video streams to a client over the ATM network for distributive VOD service, is proposed.

LSM finds the long-lasting segments which last for a long time without buffer overflow or underflow in order to lower average bit rate, bandwidth changes, and to maximize the length of segments. LSM reduces bandwidth changes and bandwidth increases, and average bit rate with a simple algorithm by eliminating the overhead produced in CBA, OBA, and SSA using a linear search for the starting point when the bandwidth changes.

Various performance metrics of bandwidth smoothing schemes are suggested, and the importance of metrics according to the service types of bandwidth reservation is discussed.

Video streams coded in MPEG-1 are employed for the experiments, and the performance results are compared and analyzed. Although the proposed LSM has a complexity O(N), the result of performance evaluation demonstrates similarities in bandwidth changes, bandwidth increases, and average segment length with respect to OBA, but shows better performances in average bit rate.

On the other hand, finding moderate buffer size and initial startup latency considering jitter and delay in the network and reducing demands for high bandwidth in the first segment are discussed.

In order to support interactive VOD service in the future, a client should transmit video data at a new frame position with the minimum tolerable delay after the completion of VCR operations. For this, the previous bandwidth smoothing algorithms, CBA and OBA in particular, are not appropriate for rescheduling. Thus, LSM is preferred for its simplicity of an algorithm as compared to CBA and OBA, which have high complexity.

In this paper a new bandwidth smoothing scheme, *Long Stretching in the Middle* (LSM) bandwidth allocation algorithm, for the distributive VOD services is discussed. Current VOD service is expected to support client's interactions in the near future. Therefore, finding methods for reducing the playback restart latency for the interactive VOD services remains for further studies.

Acknowledgements

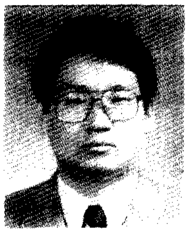
We would like to thank Jung-Min Yang for the invaluable comments and careful suggestions.

참 고 문 헌

1. The ATM Forum Technical Committee, "Audiovisual Multimedia Services: Video on Demand Specifications 1.0", af-saa-0049.000, December 1995.
2. DAVIC, "DAVIC 1.0 Specification", January 1996.
3. ISO/IEC 13818-2, "Generic Coding of Moving Pictures and Associated Audio Information: Video", January 20 1995.
4. Jean-Pierre Leduc, "Digital Moving Pictures-Coding and Transmission on ATM Networks", Elsevier Science, 1994.
5. Naohisa Ohta, "Packet Video:Modeling and Signal Processing", Artech House, 1994.
6. George Kesidis, "ATM network performance", Kluwer Academic Publishers, Chapter 6, 1996.
7. Subhabrata Sen, Jayanta K. Dey, James F. Kurose, John A. Stankovic, Don Towsley, "CBR transmission of VBR stored video", Network and Operating System Support for Digital Audio and Video, 1997.
8. Jean M. McManus, Keith W. Ross, "Video on Demand over ATM: Constant-Rate Transmission and Transport", IEEE Journal on Selected Areas in Communications, Vol. 14, No. 6, pp. 1087-1098, August 1996.
9. Jean M. McManus, Keith W. Ross, "Prerecorded VBR sources in ATM networks: Piecewise-constant-rate transmission and transport", Technical Report, Dept. of Computer Science, University of Massachusetts, September 1995.
10. J. D. Salehi, Z. -L. Zhang, J. F. Kruose and D. Towsley, "Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing", Proceeding of ACM SIGMETRICS, pp. 222-231, May 1996.
11. Wu-chi Feng and Stuart Sechrest, "Critical Bandwidth Allocation for Delivery of Compressed Video", Computer Communications, Vol. 18, No. 10, pp. 709-717, October 1995.
12. Wu-chi Feng, Farnam Jahanian, Stuart Sechrest, "An optimal bandwidth allocation strategy for the delivery of compressed prerecorded video", ACM/Springer-Verlag Multimedia Systems, Vol. 5, No. 5, pp. 297-309, 1997.
13. Wu-chi Feng and Jennifer Rexford, "Performance Evaluation of Smoothing Algorithms for Transmitting Prerecorded Variable-Bit-Rate Video", Technical Report, OSU-CISRC-3/97-TR18.
14. Junbiao Zhang and Joseph Y. Hui, "Traffic characteristics and smoothing criteria in VBR Video traffic smoothing", Proceedings of the International Conference on Multimedia Computing

and Systems, IEEE, pp. 3-11, June 1997.

15. Junbiao Zhang and Joseph Y. Hui, "Applying traffic smoothing techniques for quality of service control in VBR video transmissions", To appear in Computer Communications, special issue on building Quality of Service into Distributed Systems, 1997.
16. Junbiao Zhang, "Optimal buffering algorithms for client-server VBR Video retrievals", pp. 37, Ph. D. thesis, Rutgers University, May 1997.
17. M. Grossglauser, S. Keshav, and D. Tse, "RCBR : A simple and efficient service for multiple time-scale traffic", SIGCOMM Symposium on Communications Architectures and Protocols, ACM, pp. 219-230, August 1995.
18. The ATM Forum, "Traffic Management Specification Version 4.0", at-tm-0056. 000, April 1996.
19. Zhi-Li Zhang, James Kurose, James Salehi and Don Towsley, "Smoothing, Statistical Multiplexing and Call Admission Control for Stored Video", Technical Report UM-CS-96-29, Dept. of Computer Science, University of Massachusetts, February 1996.
20. M. W. Garrett and A. Fernandez, "Variable Bit Rate Video Bandwidth Trace Using MPEG Code", 4, November 1994.
ftp://ftp.bellcore.com/pub/vbr.video.trace/MPEG.data
21. O. Rose, "Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems", IEEE Proc. of 20th Conference on Local Computer Networks, pp. 397-406, October 1995.
ftp://ftp-info3.informatik.uni-wuerzburg.de/pub/MPEG/traces



이 재 홍(Jae-Hong Lee) 정회원
1963년 10월 9일생
1986년 2월 : 충남대학교 전자공학과 졸업(공학사)
1988년 2월 : 충남대학교 대학원 전자공학과 졸업(공학석사)
1997년 2월 : 충남대학교 대학원 컴퓨터공학과 박사과정 수료
1988년 2월~1994년 2월 : 국방과학연구소 연구원
1994년 2월~1996년 2월 : 한국인식기술 과장
※주관심분야 : VOD 서비스, 멀티미디어통신, ATM 트래픽제어

이 선 숙(Sun-Sook Lee) 정회원
1953년 10월 1일생
1979년 2월 : Bernard M. Baruch College/CUNY 회계학과 졸업(B.B.A.)
1983년 9월 : Hunter College/CUNY 전자계산학과 졸업(B.A.)
1984년 12월 : Hofstra University 대학원 전자계산학과 졸업(M.A.)
1996년 2월 : 충남대학교 대학원 컴퓨터공학과 박사과정 수료
1979년 4월~1987년 9월 : Brandvein, Kirschner & Pasternack, CPAs 및 Kim & Pecker, CPAs 회계사, 컴퓨터프로그래머
1990년 9월~1996년 3월 : 목원대학교 경리과장, 구매과장, 수서과장
※주관심분야 : 멀티미디어통신, ATM 트래픽제어, VOD 서비스

김 태 균(Tae-Kyun Kim) 정회원
1944년 1월 25일생
1971년 2월 : 서울대학교 공업교육학과 졸업(공학사)
1978년 3월 : 日本東京工業大學 大學院 物理情報工學科(공학석사)
1985년 12월 : 日本東京工業大學 大學院 物理情報工學科(공학박사)
1974년 3월~현재 : 충남대학교 공과대학 컴퓨터공학과 교수
1994년 : 충남대학교 공과대학장
※주관심분야 : 패턴인식, 영상처리, 멀티미디어, 가상현실