MPEG 동영상 전송을 위한 GOP 단위의 최소 변경 대역폭 할당 기법

곽 준 원† · 이 면 재†† · 송 하 윤††† · 박 도 순††††

요 약

비디오 서버에서 압축된 형태로 저장된 비디오 데이터를 화질의 저하 없이 클라이언트로 전송하기 위해서는 프레임당 대역폭을 일정하게 조절하는 방법이 필요하다. 이에 스무딩 기법은 프레임당 비트수가 다른 가변 비트율(VBR : Variable Bit Rate)을 가진 비디오 스트림을 일련의 고정된 전송률로 전송하는 방법이다. 본 논문에서는 비디오 서버에 저장된 비디오 스트림을 클라이언트측으로 전송할 경우 최소의 대역폭과 전송률 변화 횟수를 가지며 비디오 서버의 CPU 오버헤드를 최소화하는 스무딩 알고리즘을 제시한다. 제시된 알고리즘의 효용성을 증명하기 위해 기존의 다른 스무딩 알고리즘들과 다양한 환경에서 비교 분석하였다.

Minimum Variable Bandwidth Allocation over Group of Pictures for MPEG Video Transmission

Joon-Won Kwak† · Myoun-Jae Lee†† · Ha Yoon Song††† · Do-Soon Park††††

ABSTRACT

The transmission of prerecorded and compressed video data without degradation of picture quality requires video servers to cope with large fluctuations in bandwidth requirement. Bandwidth smoothing techniques can reduce the burst of a variable-bit rate stream by prefetching data at a series of fixed rates and simplifying the allocation of resources in the video servers and the network. In this paper, the proposed smoothing algorithm results in the optimal transmission plans for (1) the smallest bandwidth requirements, (2) the minimum number of changes in transmission rate, and (3) the minimum amount of the server process overhead. The advantages of the proposed smoothing algorithm have been verified through the comparison with the existing smoothing algorithms in diverse environments.

키워드 : 스무딩(Smoothing), MPEG, 가변 비트율(VBR), QoS

1. Introduction

With the rapid development of the Internet technology and the multimedia technology, several application areas have emerged such as Video On Demand (VOD), tele-conferencing, tele-education, virtual reality, tele-medical-diagnostics, etc. For the service in a central VOD server, a lot of video streams have soft real time constraints to transmit. In addition, the size of VOD files is usually huge to store. Thus they are usually distributed among several VOD servers. There must be at least two network links between server and client [1].

The soft real time constraint for QoS (Quality of Service) of VOD transmission is to maintain constant transmission rate and transmission delay. To cope with huge and high quality video streams transmission, several video compression techniques have been introduced and widely used which are intended to minimize the network traffic and minimize the amount of stored video.

Video compression techniques are divided into Constant Bit Rate (CBR) compressions with the constant Bits Per Frame (BPF) and Variable Bit Rate (VBR) compressions with which each frame has a different size respectively. VBR compression maintains the same quality to CBR techniques with higher compression rates while this variance of BPF may lead to a *burst*, which means an abrupt increase in the transmission size. The burst must be reduced to transmit high quality video safely on resource-restricted networks otherwise client may watch degraded video frames. Bandwidth smoothing algorithms are known as viable solution

† 준 회 원 : 홍익대학교 대학원 컴퓨터공학과
†† 준 회 원 : 홍익대학교 대학원 전자계산학과 박사수료
††† 정 회 원 : 홍익대학교 컴퓨터공학과 교수
†††† 종신회원 : 홍익대학교 컴퓨터공학과 교수
논문접수 : 2002년 7월 29일, 심사완료 : 2002년 9월 26일

to handle the high burst of VBR video transmission. The smoother resides at VOD server in order to handle a burst in cooperation with the control of network resource and the client buffer.

In this paper, we will discuss a new smoothing algorithm for VBR video stream that integrates the characteristics of the video compression technique based on the existing smoothing algorithm. We assume that the server send the video stream via network without delay while clients have enough size of buffer to contain unprocessed video streams. This is reasonable since even a thin client such as a PDA has enough size of memory currently. Among the various objective functions of smoothing algorithms [2], we choose two criteria to satisfy :

- Minimize the peak transmission rate.
- Minimize the number of changes in the transmission rate.

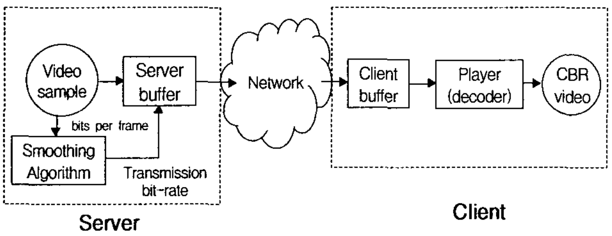
And as a premium, we earned one more benefit.

- Minimize the server process overhead.

The organization of this paper is like the followings. We will discuss existing smoothing algorithms in detail as well as basics of video compression technique specified by MPEG (Moving Pictures Experts Group) in section 2. Section 3 will present detailed description of our smoothing algorithm. Various experimental results will be presented in section 4. In the final section, we will discuss the conclusion.

2. Related Works

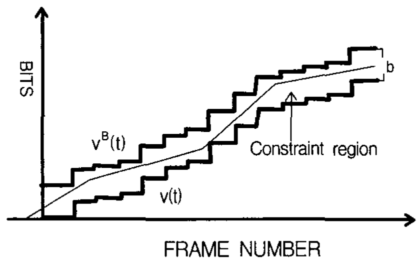
2.1 Smoothing Algorithms Basics



(Figure 1) The VOD stream transmission framework

(Figure 1) shows the framework to send a video stream from a VOD server to a client. The frames of the video compressed by the encoder temporarily are stored in a server buffer and the encoder passes the information of the frames to the smoother in the server. The smoother which is the module of smoothing algorithm calculates transmission rate in consideration of the information of buffer size in the client and the frames. And then the server sends out the video stream in the pre-calculated rate [3]. The focus of smoothing algorithm in a VOD server is reducing the burst of video streams with information of the buffer size and the frames such as BPF in order to prevent the client buffer from overflowing or underflowing.

2.2 Principles of Smoothing Algorithms



(Figure 2) Client buffer constraint

The request to a VOD server from a client includes the information of buffer size in the client. The server passes the information of the buffer size and the BPF of video stream to the smoother in the server. (Figure 2) shows the basic principles of smoothing algorithm [4]. The x-axis of this graph shows the progression of time in a unit of frame numbers passed, and y-axis shows the cumulative bits of passed frames. Variable t stands for a time in a unit of number of frames. $V(t)$ is the underflow bound of minimum transmission rate to guarantee the in-time arrival of frames to client. $V^B(t)$ is the overflow bound not to excess the size of client buffer, which will cause a loss of transmitted frames. Constant b is the size of client buffer. Constraint region is the region between $V(t)$ and $V^B(t)$ within which transmission rate can change. Following equations depict those relations more precisely.

Suppose that compressed video stream be composed of N frames. In equation (1), f_i is the size of i-th frame. $V(t)$ represents the cumulative amounts of data consumed by the client and sent by the server and denotes the underflow bound. In order to avoid the underflow of client buffer, the server must transmit enough number of frames to the client. In equation (2), $V^B(t)$ denotes the maximum cumulative amount of data received by the client at time t and is the overflow bound. In equation (3), c_i is the value of i-th change of transmission rate controlled by the smoother which value must be within the constraint region.

$$V(t) = \sum_{i=0}^{t} f_i \quad (1)$$

$$V^B(t) = V(t-1) + b \quad (2)$$

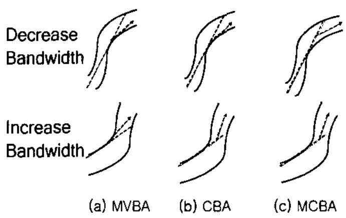
$$V(t) \le \sum_{i=0}^{t} c_i \le V^B(t) \quad (3)$$

2.3 Review of Existing Smoothing Algorithms

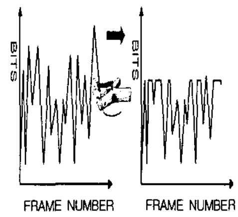
Several smoothing algorithms have been studied in [2]. As illustrated in (Figure 3), MVBA (Minimum Variability Bandwidth Allocation) [5, 6], MCBA (Minimum Changes Bandwidth Allocation) [8], and CBA(Critical Bandwidth Allocation) [7] algorithms are to minimize the number of changes in transmission rate or changes in transmission rate itself within the constraint region. While RCBS (Rate-Constrained Bandwidth Smoothing) [9] in (Figure 4) and ON-OFF (On period and Off period Smoothing Algorithm) [10] in (Figure 5) are to minimize the peak rate of transmission rate in order to meet the bandwidth requirement, PCRTT (Piecewise Constant Rate Transmission and Transport) [11] in (Figure 6) calculates the transmission rate with $V(t)$. Descriptions of existing smoothing algorithms are as follows :

- MVBA is intended to minimize the variation in the transmission rate by avoiding abrupt changes in the transmission rate. Basically, it uses backtracking mechanism to control the transmission rate with time complexity of $O(N^2)$ [5]. The improvement using queue mechanism has been made to reduce the time complexity to $O(N)$ [6].
- The number of changes in the transmission rate is minimized in MCBA algorithm [8]. This is intended to reduce the overhead of recalculation of transmission rate by network components. It requires a binary search to search a minimized number of changes.
- CBA is intended to minimize the number of changes in transmission rate when the transmission rate is increased, while the changes in transmission are minimized when the transmission rate is to be decreased, which is a mixture of MVBA and MCBA. The time complexity of CBA [7] is the similar to MVBA while it also requires a binary search tree to search the minimum number of transmission rate changes if the transmission rate are increased.
- RCBS is to minimize the utilization of client buffer with limited bandwidth [9]. This algorithm searches the frames backward to cut down the burst transmission to limited bandwidth as if it were a big chopper in (Figure 4). There is no transmission control for non-bursty frames. Larger variances have been reported both in changes in transmission rates and number of rate changes.
- ON-OFF transmits with maximum limited bandwidth denoted r at on-period but does not transmit at all at off-period. The characteristic of the algorithm is to change the length and the number between on and off period in proportion to client buffer size. A large number of changes in transmission rates are usually experienced [10].
- PCRTT calculates the average transmission rate of $V(t)$ in a predetermined time period and then rise up the offset to transmission rate in order to avoid underflow. In case of small buffer size in client, it must overflow or the time period to change transmission rate must be shorten. Therefore it will cause a larger number of changes in transmission rate [11].

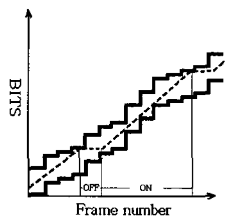
We choose MVBA, RCBS, ON-OFF and PCRTT as target algorithms to demonstrate. (Figure 7) shows the per-frame bandwidth schedule plan provided by these smoothing algorithms for 1Mbyte of client buffer size with Jurassic Park as a source.



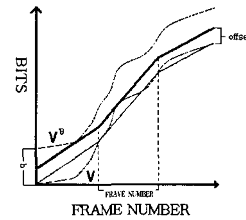
(Figure 3) MVBA, CBA and MCBA



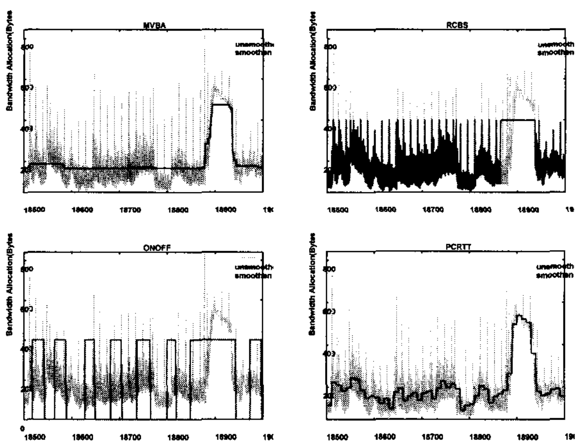
(Figure 4) RCBS



(Figure 5) ON-OFF



(Figure 6) PCRTT

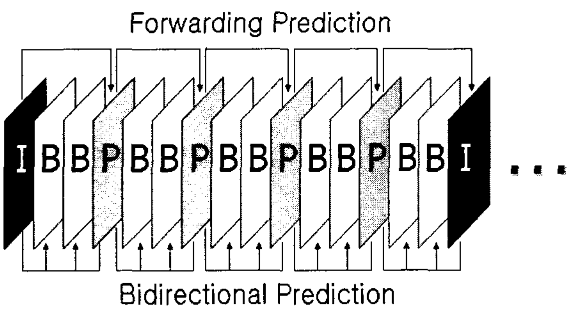


(Figure 7) Bandwidth schedule trace of existing smoothing algorithms

2.4 Video Compression Techniques

Existing smoothing algorithms deal with a unit of frame to control bandwidth allocation. Our intention is to combine the characteristics of compressed video streams to develop the smoothing technique. Among various standards of CODECs for video transmission over networks such as H.261, H.263, MPEG-1, MPEG-2, and MPEG-4, we choose MPEG-2 (Moving Pictures Expert Group) since it is widely used for various digital AV services including commercial ones.

MPEG specification [12] uses various techniques to compress video streams, which are DCT (Discrete Cosine Transform), motion compensation and Huffman code. DCT is used to remove the intra-frame redundancy and motion compensation is used to remove the inter-fame redundancy. Huffman code is one of the minimum redundancy codes.



(Figure 8) Frame patterns in a GOP

With these compression techniques, MPEG video frames are divided into three types. The first one is Intra-coded frame (I frame). It can be placed anywhere in the streams since it can be decoded independently without referencing other frames so that random access can be allowed with I frame. The second is Predictive-coded frame (P frame). P frame uses previous I frame and previous P frames to be decoded. The third is Bidirectional-coded frame (B frame), which can be coded using predictions with either past or future anchor frames (I or P frame), or both. This group of frames starting with an I frame is called as Group of Pictures (GOP). The basic pattern of GOP is in a form of IBBPBBPBBPBBPBB as shown in (Figure 8). Form the view of network transmission, frames in a GOP have inherent bursts which must be controlled.

3. MVBAG

The purpose of smoothing algorithm is to show the undegraded pictures to a client by controlling network resource allocations while sending VBR video streams. MVBA achieves minimum variability of bandwidth allocation among other smoothing algorithms. We suggest a new algorithm called MVBAG (Minimum Variability Bandwidth Allocation over Group of Pictures) that integrates GOP concept of the frame structures of MPEG and the idea of MVBA algorithm.

3.1 The Algorithm

The main idea is to execute smoothing in a unit of GOP with the fulfillment of all requirements of smoothing algorithms. For sending MPEG-2 video, the size of I frame is much considerably larger than other frames, leading larger variance in frame size. It implies that there is inherent burst in sending frames. Applying smoothing techniques in a unit of GOP, we can eliminate this inherent variance in frame size firsthand. In addition, we can achieve more efficient transmission since GOP is a required unit of MPEG video. Therefore, it is desirable to execute smoothing algorithm not for frames but for GOPs to manage the secondhand burst. With the MVBA algorithm which minimizes the variation in the transmission rate by avoiding abrupt changes in the transmission rate, MVBAG algorithm is designed to pose every positive features of MVBA and to require smaller CPU times than MVBA. (Figure 9) shows the algorithm of smoothing in a unit of GOP.

```
PROCEDURE MVBAG ( )
Initialize queue TB and T with 1 //starting GOP sequence number
(0)  REPEAT
(1)      increase t, Cmax = V(first (TB)), Cmin = V(first (T))
(2)      IF Cmax < V(t) then
(3)          REPEAT
(4)              output (Cmax), delete(first (TB))
(5)              Cmax = Constraint Bandwidth
(6)          UNTIL empty (first (TB))
(7)      delete (first (T))
(8)      ELSE IF Cmin > VB(t) then
(9)          REPEAT
(10)             output (Cmin), delete (first (T))
(11)             Cmin = V (first (T))
(12)         UNTIL empty (first (T))
(13)     delete (first (TB))
         END IF
(14)     search every convex lower bound of VB(t) in TB
(15)     insert (TB, t)
(16)     search every concave upper bound of V(t) in T
(17)     insert (T, t)
(18) UNTIL t = M
END PRCEDURE
```

(Figure 9) Pseudo-code for MVBAG procedure

In (Figure 9), line (1), *t* stands for a sequence number of

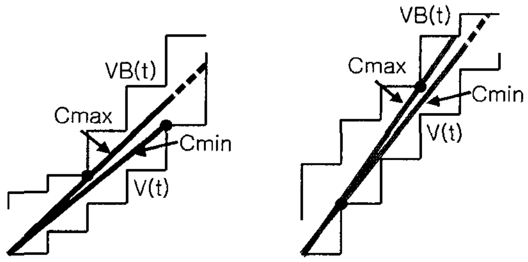
GOP, denoted as time. $V(t)$ is cumulated bits from *first* to *t-th* GOP (from time *1* to *t*) as in equation (1) and $V^B(t)$ is the sum of $V(t)$ and client buffer size as in equation (2). The alteration time of these values are held in queues *T* and *TB*, respectively. Let C_{max} be the possible maximum transmission rate in which the server can transmit. Also let C_{min} be the possible minimum transmission rate. In line (18), *M* stands for the total number of GOPs in an MPEG-2 format video source. The function first() refers the first element in a queue and it assigns initial values of C_{max} and C_{min} in line (1). C_{min} has the very initial value of $V(t)$ and C_{max} has initially allocated bandwidth.

The output() function passes the values to transmission scheduler module. The delete(), empty(), and insert() functions stand for the standard queue operations.

The convex lower bounds of $V(t)$ in line (16) mean the convex inflection times of higher transmission rates than neighboring transmission rates till time *t*, and the concave upper bounds of $V^B(t)$ in line (14) mean the concave inflection times of lower transmission rate than neighboring transmission rate till time *t*. These time values must be inserted to *T* and *TB*, respectively.

From line (14) to (17), the algorithm searches the candidate values of C_{max} and C_{min} and from line (1) to (13), it determines the values of C_{max} and C_{min} to guarantee the QoS conditions. From line (2) to (7), if C_{max} is too small to transmit frames, as shown in (a) of (Figure 10), an increased value must be recalculated at a higher rate. In lines (8) to (13), if C_{min} is large to result in buffer overflow as shown in (b) of (Figure 10), a decreased value must be recalculated at a lower rate. The increment or decrement of both rates for the next segment must be as small as possible to minimize the variance of bandwidth allocation. The algorithm repeats till the total number of GOPs processed in line (18).

(Figure 10) shows the condition of changes in transmission rate in detail. Starting the point where $V^B(t)$ hits the C_{max}, C_{max} is increased under the condition of $C_{max} < V(t)$ as shown in (a) of (Figure 10). Similarly, in (b) of (Figure 10), C_{min} is decreased after the cross point of $V(t)$ and C_{min} under the condition of $C_{min} > V^B(t)$. The increase and decrease of the maximum and the minimum transmission rates must meet the condition in equation (3), respectively.



(a) Cmax < v(t) (b) Cmin > VB(t)

(Figure 10) The situation of underflow and overflow

3.2 Time Complexity of MVBAG

N is the total number of frames in a video stream and *M* is the total number of GOPs. With *c* frames per GOP, *N* = *cM*. In this paper, we choose *c* either six or fifteen, so that *N* = *6M* or *N* = *15M*. The time complexity of MVBAG is O(*M*) while others have O(*N*).

4. Experimental Results

4.1 Video Sources Description

In order to verify the effectiveness of MVBAG in comparison with the previous algorithms, we introduced several MPEG video sources with various frame characteristics and qualities. Some movies like Star Wars and Jurassic Park have high variances in frame size and high quality pictures. News has lower size and smaller changes of frames with high quality. Music video has very impressive characteristics. The abrupt and frequent changes of scenes with several dancers moving actively in very high quality pictures

⟨Table 1⟩ Parameters of MPEG video sources

Video source	Number of Frame	Avg. BPF	Max. BPF (Kbytes)	Min. BPF (Kbytes)	Frame Std. Dev.	GOP pattern	Resolution (pixel)
Star Wars	119997	33.5	367.37	0.5	39.9	IBBPB BPBBP BBPBB	352×240
Jurassic Park	70001	15.75	286.62	0.75	10.3		
Music Video	6599	72.37	220.62	58.37	42.0		
News	22409	10.5	29.37	2.75	4.4		
Crocodile Dundee*	174483	4.5	3.5	0.01	3.41	IBBPBB	N/A
E. T.*	204664	3.5	33.37	0.02	3.23		

* The parameters for these two movies are quoted from http://www.cis.ohio-state.edu/~wuchi/Video/MPEG/index.html [2] in order to verify video sources with different parameters. These video sources are employed for fair comparison with previous researches.

will imply severe characteristics of video source to test a smoothing algorithm. Note that standard deviation (Std. Dev.) of frames size for each video source indicates the *burstiness* of each video source. Crocodile Dundee and E.T. have the similar characteristics with Star wars and Jurassic park while they have poor quality of pictures. <Table 1> summarizes the several parameters where 30 frames per second are constant for every video source.

4.2 Simulation Results

(Figure 11) shows the peak rate of transmission (Kbytes per frame), the number of changes in transmission rate, and the required CPU time to execute the smoothing algorithm for the six video sources aforementioned. The peak rate of transmission in (Figure 11) is the highest transmission rate of each video source transmission. The CPU time is average computation time of transmission rate on 100 times. The number of changes in transmission rate is determined for each video source with each algorithm. In (a) of (Figure 11), MVBAG is in the group of algorithms that accomplishes the lower peak rate. In MVBAG, Star Wars and News show the lowest peak rate than other sources. In (b) of (Figure 11), the numbers of changes in transmission rate are shown. MVBAG and MVBA outperform other smoothing algorithms in the number of changes in transmission rate. We experienced that the peak rates of bandwidth requirement are the lowest both in MVBAG and MVBA among all video sources, of course, since it was the major object function of MVBA. However, RCBS shows the highest number of changes in transmission rate among all video sources. In (c) of (Figure 11), MVBAG outperforms MVBA in CPU time consumption. Rectangular area shows the zoom in of bottom data of each graph. MVBA algorithm requires $O(N^2)$ time complexity and has been modified to achieve $O(N)$ time complexity by the introduction of queue. However, real practice showed that 50% of CPU time is consumed by the insert, manage, and delete operation of time data to queue. MVBAG works at least six times faster than MVBA in cases of Crocodile Dundee and E.T with six frames per GOP, while it sometimes shows faster execution up to eight times for other video sources. Note that MVBAG requires enough buffer size in a client in order to store video in a unit of GOP rather than that of frame.

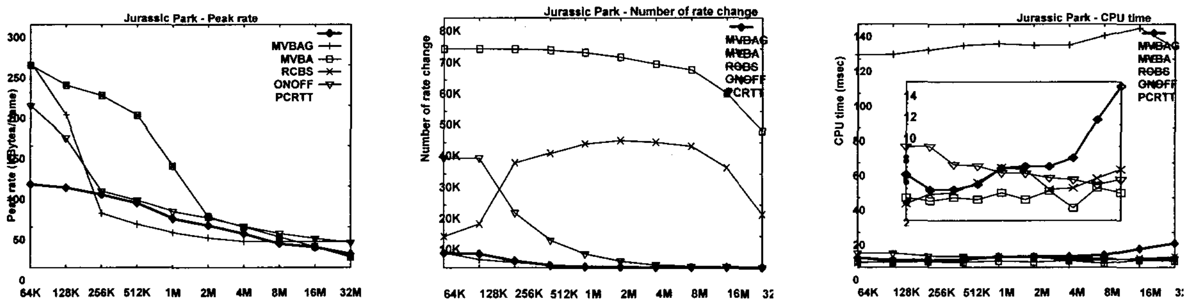
(Figure 12) shows a bandwidth schedule trace of Jurassic Park provided by MVBAG in comparison with other algorithms in (Figure 7). MVBAG shows the lowest peak transmission rate among all smoothing algorithms discussed.

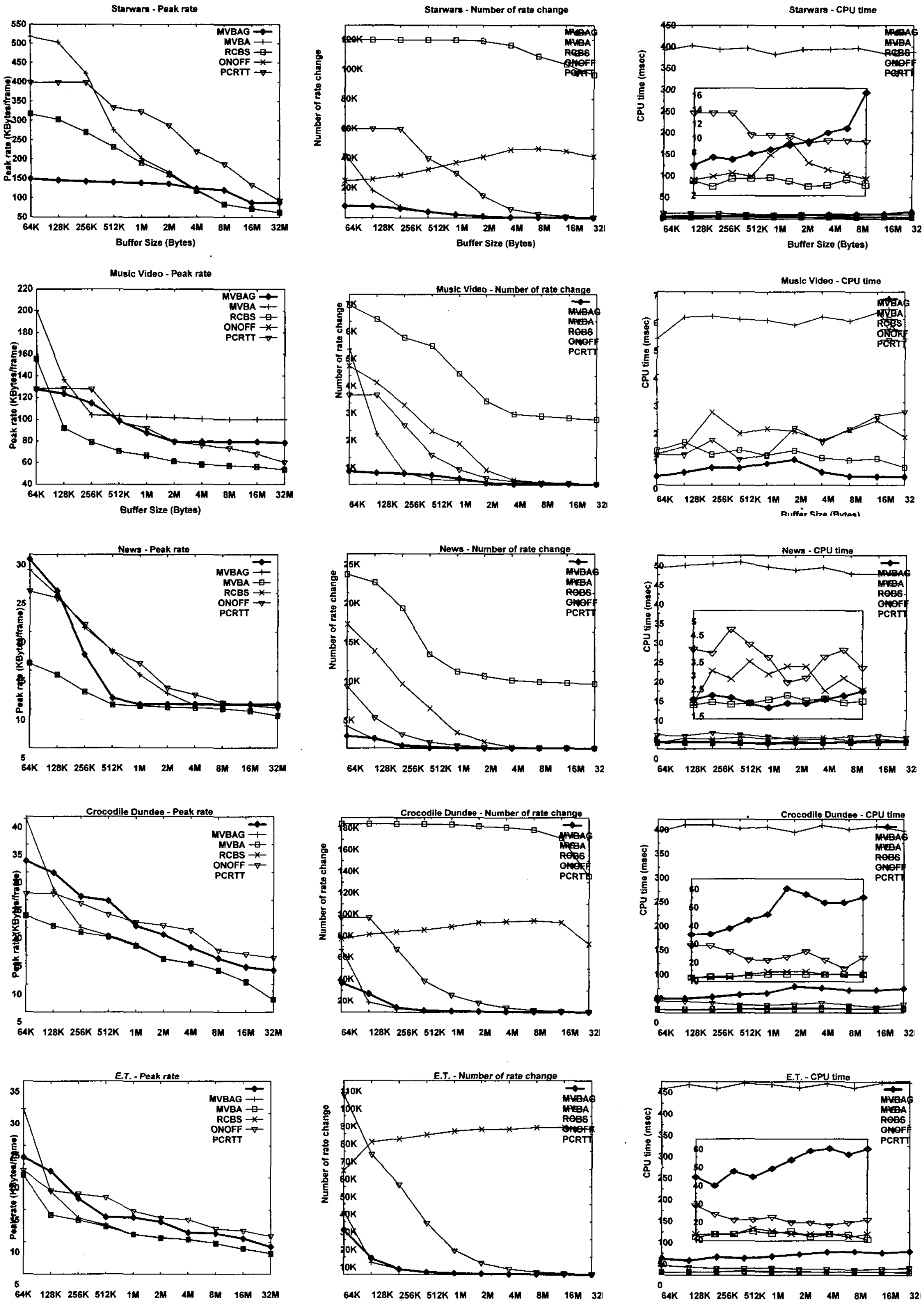
With MVBAG, as a result, we achieved the lowest peak rate and the number of changes in transmission rate across all video sources experienced, and it resides in the group of algorithms with low consumption of CPU.

5. Conclusions

We introduced MVBAG smoothing technique for transmitting VBR video streams over communication networks. We improved MVBA algorithm with the temporal frame compression characteristics of MPEG videos. The MVBAG considers GOP as basic units for smoothing while others consider frames as a unit of video transmission. The experimental result showed that MVBAG algorithm is one of the best smoothing algorithms which achieves the lowest peak transmission rates. .

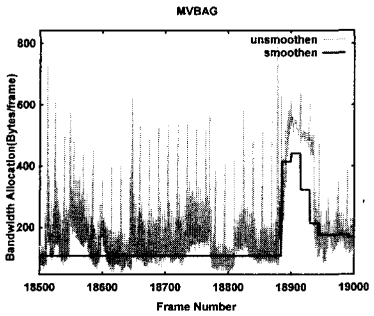
As well, the number of changes in transmission rate is one of the key parameters. The MVBAG algorithm also keeps the number of transmission rate changes as low as other algorithms. With these two advantages, MVBAG algorithm consumes CPU time as low as other smoothing algorithms as a premium which is a desirable characteristic to design a VOD server. Even though our algorithm requires more client buffers to store GOPs than other algorithms for storing frames, it is not critical since even thin clients such as PDAs have enough memory capacity.





(a) Peak rate (b) Number of rate change (c) CPU time

(Figure 11) Characteristics of MVBAG versus other smoothing algorithms



(Figure 12) Bandwidth schedule trace of MVBAG

References

[1] W. Feng, S. Sechrest, "*Critical Bandwidth Allocation for the Delivery of Compressed Prerecorded Video,*" Computer Communications, Vol.18, No.10, pp.709-717, Oct., 1995.

[2] W. Feng and J. Rexford. "*Performance evaluation of smoothing algorithms for transmitting prerecorded VBR video,*" IEEE Trans. on Multimedia, September, 1999.

[3] P. Thiran, et. al., "*Network calculus applied to optimal multimedia smoothing,*" in INFOCOM 2001.

[4] J. Zhang and J. Hui, "*Applying traffic smoothing techniques for quality of service control in VBR video transmissions,*" Computer Communications, pp375-389, April, 1998.

[5] J. D. Salehi, et. al. "*Supporting stored video : Reducing rate variability and end-to-end resource requirements through optimal smoothing,*" in Proc. of ACM SIGMETRICS, pp.222-231, May, 1996.

[6] J. D. Salehi, "*Scheduling network processing on multimedia and multiprocessor servers,*" Ph.D. dissertation, Univ. Massachusetts, Amherst, Sept., 1996.

[7] W. Feng, et. al., "*Smoothing and buffering for delivery of prerecorded compressed video,*" in Proc. of ISET/SPIE Symp. on Multimedia Comp. and Networking, pp.234-242, Feb., 1995.

[8] W. Feng, et.al., "*Optimal buffering for the delivery of compressed prerecorded video,*" Proc. of the IASTED/ISMM Int'l Conference on Networks, January, 1995.

[9] W. Feng, "*Rate-constrainted bandwidth smoothing for the delivery of stored video,*" in SPIE Multimedia Networking and Computing, 1997.

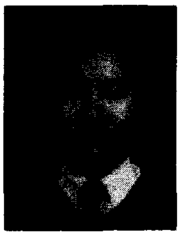
[10] Junbiao Zhang and Joseph Y. Hui, "*Traffic Characteristics and Smoothness Criteria in VBR Video Traffic Smoothing,*" in Proc. of the ICMC and Systems, Ottawa, Ontario, Canada. IEEE Computer Society, pp.3-6, June, 1997.

[11] J. M. McManus et. al., "*Video-on-Demand Over ATM : Constant-Rate Transmission and Transport,*" in IEEE Journal on selected areas in comm., Vol.14, No.6, August, 1996.

[12] D. Le Gall, "*MPEG : A video compression standard for multimedia applications,*" Communications of the ACM, Vol.34, pp.47-58, April, 1991.

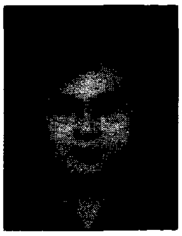
곽 준 원

e-mail : jwkwak@cs.hongik.ac.kr
2001년 홍익대학교 컴퓨터공학 졸업(학사)
2001년 현재 홍익대학교 대학원 컴퓨터공학과 재학중
관심분야 : 멀티미디어 통신, 멀티미디어 데이터 방송



이 면 재

e-mail : mjlee@cs.hongik.ac.kr
1992년 홍익대학교 전자계산학과 졸업(학사)
1994년 홍익대학교 대학원 전자계산학과 졸업(이학석사)
1994년~1999년 정원엔 시스템 연구소 근무
2000년~현재 용인 송담대학 정보통신학과 겸임교수
2001년~2002년 코리아 홈넷 수석 연구원
2002년~현재 아라마루 수석 연구원, 홍익대학교 대학원 전자계산학과 박사수료
관심분야 : 멀티미디어 통신, 멀티미디어 데이터 방송, 설계자동화



송 하 윤

e-mail : song@cs.hongik.ac.kr
1991년 서울대학교 계산통계학 졸업(학사)
1993년 서울대학교 대학원 전산과학 졸업(공학석사)
1995년~2001년 Univ. of California, Los Angeles Computer Science
2001년~현재 홍익대학교 컴퓨터공학과 교수
관심분야 : 멀티미디어 통신, 멀티미디어 데이터 방송, 홈네트워크, 위성통신, 분산병렬시뮬레이션

박 도 순

e-mail : dspark@cs.hongik.ac.kr
1978년 서울대학교 전자공학 졸업(학사)
1980년 한국과학기술원 대학원 전자공학 졸업(공학석사)
1988년 고려대학교 대학원 전자계산 졸업(공학박사)
1980년~1983년 국방과학연구소 선임연구원
1983년~현재 홍익대학교 컴퓨터공학과 교수
관심분야 : 컴퓨터구조, 설계자동화