

# 인터넷상에서 멀티미디어 서비스를 위한 전송률 조절 알고리즘

이면재\*, 박도순  
홍익대학교 전자계산학과  
{mjlee, dspark}cs.hongik.ac.kr

## A Transmission Rate Control Algorithm for Multimedia Service in Internet

Myoun-Jae Lee\*, Do-Soon Park  
Dept. of Computer Science, Hongik University

### 요 약

스무딩은 가변 비트율로 저장된 비디오 데이터를 클라이언트로 전송할 때 일련의 고정 비트율로 전송할 수 있도록 전송 계획을 세우는 것이다. 이러한 스무딩 알고리즘에는 CBA, MCBA, MVBA, PCRTT, e-PCRTT 등이 있다. 특히, PCRTT 알고리즘의 문제점을 개선한 e-PCRTT 알고리즘은 전송률 변화 횟수가 주어지고 구간의 크기가 고정적인 특징을 갖고 있어 전송률 변화 횟수, 첨두 전송률, 버퍼 이용률 등의 평가 요소들이 증가될 수 있다. 따라서, 본 논문에서는 e-PCRTT 알고리즘의 문제점을 해결하기 위해 전송률 변화 횟수의 제한이 없고 구간의 크기가 가변적인 스무딩 알고리즘을 제안한다. 제안 알고리즘의 성능 평가를 위해 3개의 비디오 소스를 사용하여 전송률 변화 횟수, 첨두 전송률, 그리고 버퍼 이용률을 비교한다. 제안 알고리즘은 전송률 변화 횟수, 첨두 전송률은 e-PCRTT 알고리즘 보다 우수하고, 버퍼 이용률 비교에서는 비슷한 결과를 보인다.

### 1. 서론

디지털 도서관과 VoD(Video on Demand) 서비스와 같은 멀티미디어 응용에 사용되는 비디오 데이터는 네트워크 자원과 저장 장치의 효율성을 위해 압축되어 저장된다[1]. 이러한 압축방법에는 프레임을 구성하는 비트 수가 동일한 고정 비트율(Constant Bit Rate) 방식과 프레임을 구성하는 비트 수가 동일하지 않은 가변 비트율(Variable Bit Rate) 방식이 있다. 가변 비트율 방식은 고정 비트율 방식에 비해 비디오 화질과 압축률이 우수하지만, 프레임을 구성하는 비트 수의 차이가 심하므로 이것을 그대로 전송하면 QoS(Quality of Service)를 보장하기 위해서 전송률을 급격히 증가시켜야 되는 버스트(Burst) 현상이 발생할 수 있다[2]. 따라서, 비디오 서버는 버스트를 피하기 위해 가변 비트율로 저장된 비디오 데이터를 일련의 고정 비트율로 전송할 수 있도록 전송 계획을 세워야 하는데, 이 기법을 스무딩(Smoothing)이라고 한다[3]. 이러한 스무딩 기법을 위한 알고리즘에는 CBA(Critical Bandwidth Allocation), MCBA(Minimum Changes Bandwidth Allocation), MVBA(Minimum Variability Bandwidth Allocation), PCRTT

(Piecewise Constant Rate Transmission and Transport) 등이 있다. 이 중 전송률 변화 횟수가 주어지는 PCRTT(Piecewise Constant Rate Transmission and Transport) 알고리즘은 전송률 변화 주기가 일정하고 클라이언트 버퍼 크기를 최소화 하는 것을 목적[4,5]으로 하지만, 모든 구간에 옴셋 값을 더하므로 불필요하게 버퍼 크기가 커질 수 있다. 특히, PCRTT 알고리즘의 문제점을 개선한 e-PCRTT(enhanced-PCRTT) 알고리즘은 PCRTT 알고리즘에서와 같이 전송률 변화 횟수에 제한이 있고 구간의 크기가 일정한 특징을 갖고 있기 때문에, 전송률 변화 횟수, 첨두 전송률, 버퍼 이용률 등의 스무딩 알고리즘의 평가 요소들이 증가될 수 있다.

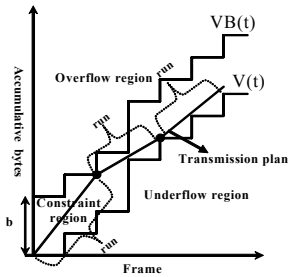
본 논문에서는 e-PCRTT의 문제점을 해결하기 위해 전송률 변화 횟수의 제한이 없고 구간의 크기가 가변적인 스무딩 알고리즘을 제안한다. 제안 알고리즘과 e-PCRTT 알고리즘의 성능 평가를 위해 3개의 비디오 소스를 사용하여 전송률 변화 횟수, 첨두 전송률, 버퍼 이용률을 비교 하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 설명하고, 3장에서는 제안 알고리즘을 기술한다. 4

장에서는 실험 결과를 비교 분석하고, 5장에서는 결론 및 추후 연구 방향을 기술한다.

2. 관련 연구

스무딩 기법은 가변 비트율로 저장된 비디오 스트림을 전송할 때 버스트를 줄이기 위해 일련의 고정 비트율로 전송하는 방법이다. [그림 1]은 스무딩 알고리즘의 원리이다.



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

$$VB(t) = b + \sum_{i=0}^t f_i \dots(2)$$

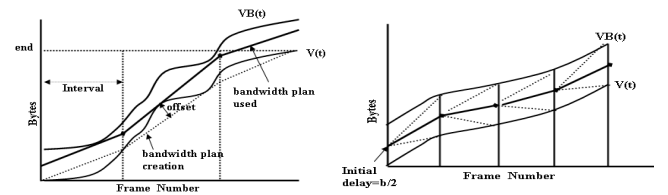
[그림 1] 스무딩 알고리즘의 원리

[그림 1]의 V(t)는 언더플로우 경계선으로, 서버가 이 경계선보다 낮은 전송률로 프레임을 전송하면 클라이언트에서 언더플로우가 발생되어 QoS를 보장할 수 없다. 식(1)의 V(t)는 언더플로우 경계선으로 프레임 0부터 프레임 t까지의 누적된 바이트 수이며,  $f_i$ 는 프레임 i의 바이트 수이다.

VB(t)는 오버플로우 경계선으로 식(2)와 같이 언더플로우 경계선에 버퍼 크기 b를 더한 값으로 표현된다. 서버가 오버플로우 경계선 보다 큰 전송률로 프레임을 전송하면 클라이언트에서 오버플로우가 발생되어 QoS를 보장할 수 없다. 따라서, 서버에서 계산된 전송률이 QoS를 만족하려면 언더플로우 경계선과 오버플로우 경계선 영역 내에 있어야 하는데, 이 영역을 제한 영역(Constraint region)이라고 하며, 이 제한 영역에서 동일한 전송률로 보낼 수 있는 프레임들의 구간이 런(Run)이다.

제한 영역을 만족하고, 전송률 변화 횟수, 첨두 전송률, 버퍼 이용률등의 특정 요소를 최적화하는 목적으로 다양한 스무딩 알고리즘이 연구되어 왔다. CBA 알고리즘은 전송률 증가 횟수를 최소화 하는 것이 목적이며, MCBA 알고리즘의 목적은 전송률 변화 횟수의 최소화이고, MVBA 알고리즘의 목적은 전송률 변화량을 최소화하는 것이다. 또한, PCRTT 알고리즘의 목적은 일정한 전송률 변화 주기와 클라이언트 버퍼 크기를 최소화 하는 것이다[4]. 특히, 전송률 변화 횟수가 주어지는 PCRTT 알고리즘은

전체 비디오 스트림을 구성하는 프레임 개수를 전송률 변화 횟수로 나눈 몫을 1개의 구간 크기로 설정한다. 이 구간 크기로 비디오 스트림을 분할하고 각 구간의 시작 프레임에서 구간의 끝 프레임과 언더플로우 경계선인 V(t)와 교차하는 지점을 연결하는데, 이 연결선이 해당 구간의 임시 전송률이다. 이와같이 모든 구간의 임시 전송률을 계산하고 언더플로우가 가장 크게 발생된 프레임이 속한 구간의 프레임들이 언더플로우를 피할 수 있는 최소의 오프셋 값을 구하고 이 오프셋 값을 모든 임시 전송률에 더해 각 구간의 전송률을 설정한다. PCRTT 알고리즘은 특정 구간만 언더플로우가 심하게 발생되었을 때에도 모든 구간에 오프셋 값을 더하므로 버퍼 이용률과 비디오 데이터의 전송에 요구되는 버퍼 크기가 커질 수 있다. 이 문제점을 개선하기 위하여 e-PCRTT 알고리즘은 각 구간의 언더플로우가 발생되지 않을 최소 전송률과 오버플로우가 발생되지 않을 최대 전송률의 평균을 해당 구간의 전송률로 설정한다. [그림 2]는 PCRTT 알고리즘의 전송률 조절 과정이고[6], [그림 3]은 e-PCRTT 알고리즘의 전송률 조절 과정이다[6].



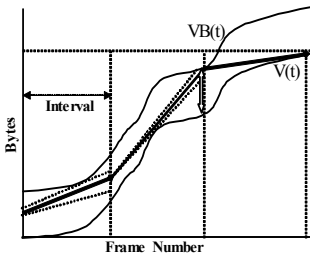
[그림 2] PCRTT 알고리즘의 전송률 조절 과정 [그림 3] e-PCRTT 알고리즘의 전송률 조절 과정

3. 제안 알고리즘

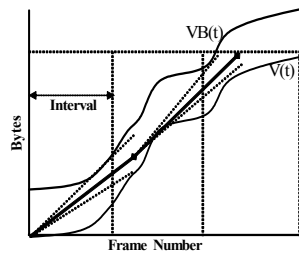
구간마다 전송률이 변화되어야 하는 e-PCRTT 알고리즘은 불필요하게 전송률 변화 횟수가 증가 될 수 있다. 즉, [그림 3]의 2번째 구간과 3번째 구간은 동일한 전송률로 설정될 수 있으나 구간이 다르므로 전송률을 변경해야 한다. 또한, 제한된 전송률 변화 횟수는 구간 크기, 전송률 변화 횟수, 첨두 전송률, 전송률 변화량, 버퍼 이용률 등에 영향을 준다. [그림 4]는 전송률 변화 횟수 제한이 3인 경우에 e-PCRTT 알고리즘의 전송률 조절 과정을 보여주는데, 구간 크기가 고정적이고 구간 당 1개의 전송률로 보내야 하는 알고리즘의 특징 때문에 2번째 구간에서 급격히 전송률이 변화되어 첨두 전송률과 전송률 변화량이 커진 것을 볼 수 있다.

본 논문에서는 앞에서 언급한 문제들을 해결하기

위해, 전송률 변화 횟수의 제한이 없고 구간 크기가 가변적인 알고리즘을 제안한다.



[그림 4] e-PCRTT 알고리즘의 전송률 조절 문제점



[그림 5] 제안 알고리즘에 의한 [그림 4]의 전송률 조절 예

[표 1]은 제안 알고리즘이다.  $t$ 는 프레임 번호,  $ts$ 는 런의 시작 프레임,  $N$ 은 비디오 스트림을 구성하는 프레임 개수,  $C_{max}$ 는 런의 시작 프레임부터 현재 검색되고 있는 프레임  $t$ 까지 QoS를 보장하는 전송률 중 최대 전송률,  $C_{min}$ 은 최소 전송률이다. 또한,  $Out(ts)$ 는 프레임 0부터 프레임  $ts$ 까지의 프레임들을 서버에서 계산된 전송률로 클라이언트에게 보내는 경우에 누적되는 바이트 수이다. (1)의  $mb_{(t)}$ 는 런의 시작 프레임부터 현재 검색중인 프레임  $t$ 의 오버플로우 경계선과 언더플로우 경계선의 평균 전송률이다. (2)에서 (5)까지는  $mb_{(t)}$ 가 QoS를 보장하는 범위에 있지 않다면 런의 시작 프레임부터 프레임  $t-1$ 까지의 전송률은  $mb_{(t-1)}$ 로 설정되고 다음 런의 시작 프레임이  $(t-1)$  프레임으로 설정되는 부분이다.  $QoS\_Check(mb_{(t)})$ 는  $mb_{(t)}$ 가 QoS를 만족하는 여부를 판단하는 함수이다. (6)에서 (7)까지는  $C_{max}$ 와  $C_{min}$ 을 계산하는 부분이다.

```

Initialize  $t=0, ts=0, C_{max}, C_{min}$ 
REPEAT
(1)  $mb_{(t)} = ((V(t)+VB(t))/2.0) - (Out(ts))/(t-ts)$ 
(2) IF NOT(QoS_Check( $mb_{(t)}$ ))
(3)   build the run as  $mb_{(t-1)}$  from  $ts$  to  $t-1$  frame
(4)    $ts=t-1$ ; //determine starting frame of next run
(5)   ENDIF
(6)   Compute  $C_{max}, C_{min}$ 
(7) UNTIL  $t = N$ 
    
```

[표 1] 제안 알고리즘

[그림 5]는 제안 알고리즘에 의한 [그림 4]의 전송률 조절 예이다. 제안 알고리즘은 구간 내에서 전송률

이 변화되어 전송률 변화량, 첨두 전송률등이 불필요하게 커지지 않는다.

#### 4. 실험 결과

제안 알고리즘과 e-PCRTT 알고리즘은 C언어로 구현하였다. [표 2]는 MPEG-2로 저장된 비디오 소스들의 파라미터로써, 비디오 스트림을 구성하는 프레임의 바이트 수를 다운로드[7,8] 하여 계산하였다.

Length는 비디오 재생 시간, Ave Frame Size는 비디오 스트림의 각 프레임을 구성하는 바이트 수의 평균, Std.Dev는 프레임을 구성하는 바이트 수의 표준 편차로서 이 값이 클수록 프레임을 구성하는 바이트 수의 변화가 심하다. 1993 Final Four는 프레임 당 바이트 수의 변화가 가장 심한 비디오 소스이고 Seminar는 프레임 당 바이트 수의 변화가 아주 적은 비디오 소스이다.

Video Clip Name	Length (min)	Ave Frame Size (KB)	Std.Dev (KB)
Crocodile Dundee	94	2.59	2.281
Seminar	63	2.07	0.578
1993 Final Four	41	3.95	4.041

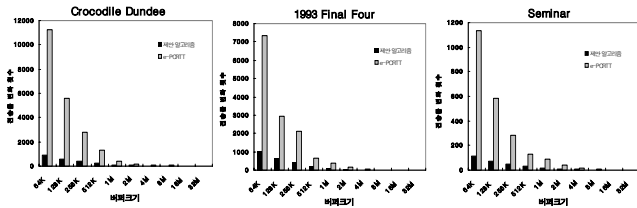
[표 2] MPEG 비디오 소스의 파라미터들

전송률 변화 횟수의 제한이 있고 구간의 크기가 일정한 e-PCRTT 알고리즘과 제안 알고리즘의 성능 평가를 위해 전송률 변화 횟수, 첨두 전송률, 버퍼 이용률을 비교하였다. e-PCRTT에서 구간 크기는 언더플로우 또는 오버플로우가 발생되지 않으면서 구성할 수 있는 구간 크기의 최대 값으로 설정하였다. 클라이언트 버퍼 크기는 64K부터 32M로 다양하게 설정하였다.

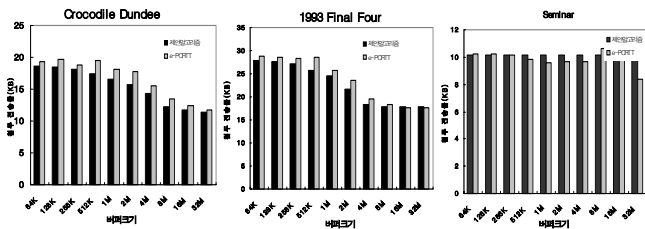
[그림 6]은 e-PCRTT 알고리즘과 제안 알고리즘의 전송률 변화 횟수 비교이다. 버퍼 크기가 작은 경우에 제안 알고리즘의 전송률 변화 횟수가 e-PCRTT보다 적는데, 이것은 e-PCRTT 알고리즘에서 주어진 버퍼 크기를 만족하고 언더플로우 또는 오버플로우가 발생되지 않으면서 구성할 수 있는 최대 구간의 크기가 작게 되어 제안 알고리즘보다 구간 개수가 많기 때문이다.

[그림 7]은 e-PCRTT 알고리즘과 제안 알고리즘의 첨두 전송률 비교이다. 첨두 전송률은 서버에서 계산된 전송률 중 가장 높은 전송률이다. 프레임을 구성하는 바이트 수의 표준 편차가 심한 Crocodile

Dundee와 1993 Final Four 비디오 소스에서는 제안 알고리즘의 첨두 전송률이 더 낮고 Seminar 비디오 소스에서는 비슷하다. 이것은 프레임을 구성하는 표준 편차가 심한 비디오 소스의 경우에 e-PCRTT 알고리즘에서 구간의 시작 프레임과 끝 프레임의 프레임을 구성하는 바이트 수의 차이가 심해 전송률을 급격하게 증가시켜야 하기 때문이라고 판단된다.

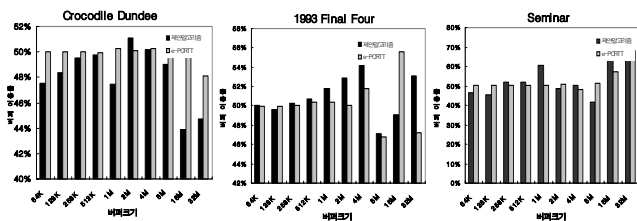


[그림 6] 전송률 변화 횟수 비교



[그림 7] 첨두 전송률 비교

[그림 8]은 e-PCRTT 알고리즘과 제안 알고리즘의 버퍼 이용률 비교이다. 클라이언트 버퍼 이용률은 다른 비디오 스트림 또는 응용 프로그램들이 버퍼를 공유하므로 낮을 수록 좋다[4]. 제안 알고리즘과 e-PCRTT의 버퍼 이용률은 비슷하다. 또한, 제안 알고리즘의 버퍼 이용률은 비디오 소스와 버퍼크기에 따라 차이가 있지만 각 구간의 언더플로우가 발생되지 않을 최소 전송률과 오버플로우가 발생되지 않을 최대 전송률의 평균을 해당 구간의 전송률로 설정하는 e-PCRTT 알고리즘의 버퍼 이용률은 약 50%로 일정하다.



[그림 8] 버퍼 이용률 비교

### 5.결론 및 추후 연구 방향

본 논문에서는 전송률 변화 횟수가 주어지지 않고 구간의 크기가 가변적인 알고리즘을 제안하고, 전송률 변화 횟수의 제한이 있고 구간의 크기가 고정적인 e-PCRTT 알고리즘과 제안 알고리즘의 성능을 비교 분석하였다. 제안 알고리즘은 e-PCRTT 알고리즘보다 전송률 변화 횟수, 첨두 전송률 비교에서는 우수하고, 버퍼 이용률 비교에서는 비슷한 결과를 보인다. 추후에는 다양한 비디오 소스로써 기존의 여러 스무딩 알고리즘과 제안 알고리즘의 성능을 비교 분석할 예정이다.

### 참고문헌

[1]Wu-chi Feng, "Rate-constrained bandwidth smoothing for the delivery of stored video", SPIE Multimedia Computing and Networking, 1997.  
 [2]Han-Chieh Chao,C.L.Hung,Y.C.Chang,"Efficient Changes and Variability Bandwidth Allocation for VBR Media Streams",IEEE,2001.  
 [3]D.Le Gall,"MPEG:A Video compression standard for multimedia applications",Communications of the ACM,vol.34,pp.47-58,April 1991.  
 [4]W. Feng and J. Rexford. "Performance evaluation of smoothing algorithms for transmitting prerecorded VBR video", IEEE Trans. on Multimedia, September 1999.  
 [5]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.  
 [6]Ofer Hadar Reuven Cohen, "PCRTT Enhancement for Off-Line Video Smoothing",The Journal of Real Time Imaging. Vol. 7, No. 3, pp. 301-314, June 2001.  
 [7]http://www.cis.ohio-state.edu/~wuchi  
 [8]Wu-Chi Feng, "VIDEO-ON-DEMAND SERVICES :EFFICIENT TRANSPORTATION AND DECOMPOSITION OF VARIABLE BIT RATE VIDEO", PH.D, Computer Science and Engineering, 1996.