

# 개선된 e-PCRTT 알고리즘들의 성능 분석

이면재\*, 박도순

남서울대학교 멀티미디어학과\*, 홍익대학교 컴퓨터공학과  
mjlee@nsu.ac.kr, dspark@hongik.ac.kr

## Performance Evaluation of Improved e-PCRTT Algorithms

Myoun-Jae Lee\*, Do-Soon Park

Dept. of Multimedia Namseoul University\*,  
Dept. of Computer Science Hongik University

### 요 약

적용형 대역폭 할당 방법에서는 가변 비트율의 비디오 데이터에 대해 스무딩 알고리즘을 사용하여 전송 계획을 세우고, 이 계획에 따라 데이터를 네트워크 트래픽을 고려하여 전송한다. 그러나, 새로운 구간에서 높은 전송률이 요구될 수 있는 개선된 e-PCRTT 알고리즘[4]을 이 방법에서의 전송 계획으로 그대로 사용하려는 경우 가용 전송률이 낮은 경우에는 폐기되는 프레임의 양이 많아질 수 있다. 이를 개선한 알고리즘[5]에서는 언더플로우 경계선과 오버플로우 경계선의 중간에서 전송률이 변화되지만, 전송률의 증가가 필요한 경우에는 전송률 증가 양을 최소화 한다. 본 논문에서는, 개선된 e-PCRTT 알고리즘[4]과 이를 다시 개선한 알고리즘[5]의 성능을 평가하기 위하여 E.T.90 비디오 데이터를 사용하여, 침투 전송률, 평균 전송률 증가량, 폐기되는 프레임 양을 비교하였다.

### 1. 서론

디지털 도서관과 VoD(Video of Demand) 서비스에 사용되는 비디오 데이터는 네트워크 자원과 저장 장치의 효율성을 위해 가변 비트율 방법으로 압축되어 저장된다[1]. 그러나, 이 방법으로 저장된 비디오 데이터는 프레임을 구성하는 비트 수의 차이가 심하므로 이것을 그대로 전송하면 QoS를 보장하기 위해서 전송률을 급격히 증가시켜야 되는 버스트(Burst) 현상이 발생할 수 있다[2]. 이 문제를 해결하기 위하여 스무딩(Smoothing), 적용형 비디오 전송(Adaptable Video Transmission), 적용형 대역폭 할당(Adaptable Bandwidth Allocation) 방법 등[1,2]이 사용된다. 스무딩은 네트워크의 트래픽을 고려하지 않고 전송 계획을 수립하여 비디오 스트림을 전송하는 방법이다. 이 방법을 위한 알고리즘에는 CBA(Critical Bandwidth Allocation)[3], MCBA(Minimum Changes Bandwidth Allocation)[7], MVBA(Minimum Variability Bandwidth Allocation)[8], e-PCRTT(enhanced-Piecewise Constant Rate Transmission and Transport)[9] 등이 있다. 이 방법은 서버에서 세운 전송 계획대로 네트워크 자원들이 확보되는 환경에서만 침투 전송률, 전송률 변화량 등의 특정 요소에 최적화 된 전송을 할 수 있다. 적용형 비디오 전송은 네트워크 트래픽을 고려하지만 전송 계획을 세우지 않고 프레임들을 전송하는 방법으로, 네트워크 트래픽이 유동적인 환경에 사용될 수 있지만 특정 요소에 대한 최적화된 전송을 할 수 없다. 적용형 대역폭 할당은 스무딩 알고리즘을 사용하여

전송 계획을 수립하고, 전송 계획에서 요구되는 전송률이 가용 전송률 보다 큰 경우에는 가용 전송률을 만족시킬 때까지 일부 프레임을 폐기하여 재생률을 감소시키는 형태로 네트워크 트래픽을 반영한다. 이 방법에는 변경된 CBA(Modified CBA)와 적용형 CBA(Adaptable CBA)가 있으며, 네트워크의 트래픽이 유동적인 환경에서 특정 요소를 최적화하면서 전송할 수 있다[1,2]. 그런데, 이 방법에서의 폐기되는 프레임의 양은 전송률 증가량이 갑자기 증가되는 경우 많아질 수 있다. 즉 가용 전송률이 낮고 새로운 구간에서 전송률이 급격하게 증가되는 경우 가용 전송률을 만족시키기 위해 새로운 구간에서 많은 프레임이 폐기될 수 있다. 이와 같은 상황은 전송률 증가 양을 고려하지 않는 e-PCRTT를 개선한 알고리즘[4]를 적용형 대역폭 할당 방법에서의 전송 계획으로 사용하려는 경우에 발생할 수 있다. 이를 개선한 알고리즘[5]에서는 언더플로우 경계선과 오버플로우 경계선의 중간에서 전송률이 변화되지만, 전송률 증가가 요구되는 경우에 전송률 증가 양을 최소화 하여 연속적인 두 구간의 전송률이 완만하게 변화될 수 있게 한다.

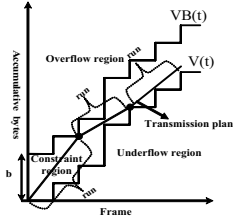
본 논문에서는, e-PCRTT를 개선한 알고리즘[4]와 [4]를 다시 개선한 알고리즘[5]의 성능을 네트워크 트래픽을 고려하지 않는 환경과 네트워크 트래픽을 고려한 환경에서 각각 평가한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 설명하고, 3장에서는 성능 평가를 기술하고, 4장에서는 결론 및 추후 연구 방향을 기술한다.

2. 관련 연구

2.1 스무딩

스무딩 기법은 가변 비트율로 저장된 비디오 스트림을 전송할 때 버스트를 줄이기 위해 일련의 고정 비트율로 전송하는 방법이다. (그림 1)은 스무딩 기법의 원리이다[3].



(그림 1) 스무딩 기법의 원리

식 (1)에서  $t$ 는 프레임 번호를 나타내며,  $f_i$ 는 프레임  $i$ 의 바이트 수를 의미한다.  $V(t)$ 는 언더플로우 경계선으로 프레임 0부터 프레임  $t$ 까지의 누적된 바이트 수로 표현되며 이 경계선보다 낮은 전송률로 프레임을 전송하면 클라이언트에서 언더플로우가 발생된다. 식 (2)의  $VB(t)$ 는 오버플로우 경계선으로 언더플로우 경계선에 버퍼 크기  $b$ 를 더한 값으로 표현된다. 이 경계선보다 높은 전송률로 프레임을 전송하면 클라이언트에서 오버플로우가 발생된다. 따라서, 서버에서 세운 전송률이 QoS를 만족하려면 언더플로우 경계선과 오버플로우 경계선 영역 내에 있어야 하는데, 이 영역을 제한 영역(Constraint region)이라고 하며, 이 영역에서 동일한 전송률로 전송할 수 있는 프레임들의 구간이 런(Run)이다.

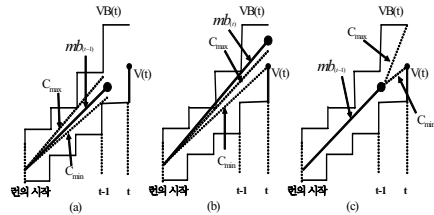
(그림 1)의 원리를 바탕으로 전송률 변화 횟수, 침투 전송률, 버퍼 이용률 등의 특정 요소를 최적화하는 목적으로 다양한 스무딩 알고리즘이 연구되어 왔다. CBA 알고리즘[3]은 전송률 증가 횟수의 최소화가 목적이며, MCBA 알고리즘[6]은 전송률 변화 횟수의 최소화가 목적이며, MVBA 알고리즘[8]은 전송률 변화량을 최소화하는 것이 목적이다. 구현의 용이함과 네트워크의 전송률 예약에 대한 통신 비용을 줄이려는 목적을 갖는 CRTT(Constant Rate Transmission and Transport) 알고리즘[6]에서는 런의 개수가 1개이어서 클라이언트 버퍼 크기와 초기 지연 시간이 크고, 침투 전송률의 이용률이 낮을 수 있다[6]. 이러한 단점을 개선하여 PCRTT(Piecewise Constant Rate Transmission and Transport) 방법[6,9]에서는 전송률 변화 횟수 즉 구간의 개수를 설정함으로써 CRTT 방법보다 버퍼 크기를 줄였지만 언더플로우를 피하기 위해 모든 구간에 동일한 윗값을 적용하므로 버퍼 크기가 아직도 커질 수 있다[9]. e-PCRTT 알고리즘[9]에서는 버퍼 크기가 커질 수 있는 PCRTT 방법의 단점을 개선하기 위하여 초기

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

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

지연 시간 동안 클라이언트 버퍼에 버퍼 크기의 1/2에 해당하는 바이트 수를 미리 저장하고 각 구간에서 언더플로우가 발생되지 않을 최소 전송률과 오버플로우가 발생되지 않을 최대 전송률의 평균을 해당 구간의 전송률로 설정한다. 그러나, e-PCRTT 알고리즘은 PCRTT 알고리즘과 같이 전송률 변화 횟수에 제한을 갖고 있고 구간의 크기가 일정하여서 불필요한 전송률 변화가 발생되거나 요구되는 클라이언트 버퍼 크기가 커질 수 있다. 이를 개선한 전송률 변화 횟수에 제한이 없고 구간의 크기가 가변적인 스무딩 알고리즘(개선된 e-PCRTT 알고리즘 1)[4]에서는 (그림 2)의 방법으로 런을 설정한다.  $C_{max}$ 는 런의 시작 프레임부터 현재 검색하고 있는 프레임까지 QoS를 보장하는 전송률 중에서 최대 전송률이고  $C_{min}$ 은 최소 전송률이다.

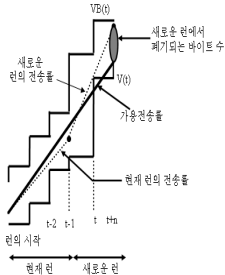
$mb_{(t-1)}$ 는 현재 검색중인 프레임 ( $t-1$ )에 의해 생성되는 전송률이다. (그림 2) (a)는 검색되는 프레임 ( $t-1$ )에서 설정하려는 전송률이 QoS를 만족하기 때문에 다음 프레임  $t$ 를 검색해야하는 상황이다. (그림 2) (b)는 프레임  $t$ 에서 설정하려는 전송률  $mb_{(t)}$ 가 QoS를 만족하지 않아서 프레임  $t$ 는 현재 런이 될 수 없음을 보여준다. 그래서, (그림 2) (c)에서와 같이 런의 시작 프레임부터 프레임 ( $t-1$ )까지의 프레임들을 프레임 ( $t-1$ )에서의 전송률, 즉  $mb_{(t-1)}$ 로 설정하여 현재 런을 생성한다.



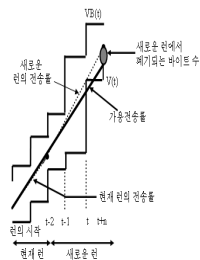
(그림 2) 개선된 e-PCRTT 알고리즘 1

개선된 e-PCRTT 알고리즘 1에서는 (그림 2)의 과정으로 현재 런을 구하는데, 전송률 증가량을 고려하지 않고 새로운 런을 구하므로 새로운 구간에서 급격하게 높은 전송률이 요구될 수 있다. 그래서, 이 알고리즘을 적응형 대역폭 방법에서의 전송 계획으로 사용하는 경우에 많은 프레임이 폐기되어야만 하는 상황이 발생할 수 있다. (그림 3)은 이러한 상황의 예인데, 현재 런에서는 가용 전송률이 런의 전송률보다 커서 폐기되는 프레임이 없지만 새로운 런에서는 가용 전송률이 런의 전송률보다 작아서 프레임들이 폐기되어야 한다. 그러나 (그림 4)에서와 같이 현재 런을 구성하는 프레임들 중에서 프레임 ( $t-2$ )까지를 현재 런으로 완성하여

현재 런과 새로운 런의 전송률 증가를 완만하게 변화시킨다면, 가용 전송률을 만족시키기 위해 폐기시켜야 되는 프레임 양이 (그림 3)에서 보다 적어질 수 있다. 이와 같은 방법으로 알고리즘[5](개선된 e-PCRTT 알고리즘 2)에서는 전송률의 증가가 요구되는 경우에 전송률을 완만하게 변화시킨다.



(그림 3) 개선된 e-PCRTT 알고리즘 1에서의 폐기되는 프레임 양의 예

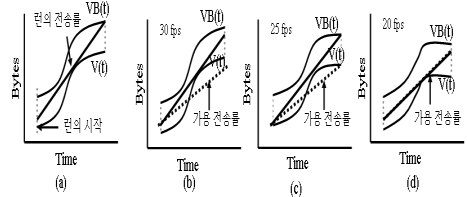


(그림 4) 폐기되는 프레임 양을 줄이기 위한 방법의 예

2.2 적응형 대역폭 할당 방법

적응형 대역폭 할당은 비디오 데이터에 대한 전송 계획을 수립한 후에 네트워크 트래픽을 고려하여 전송하는 방법이다. 즉, 스무딩 알고리즘에 의한 전송 계획에서 요구되는 전송률이 네트워크의 가용 전송률보다 큰 경우에는 가용 전송률을 만족시킬 때까지 전송하려는 프레임들 중에서 일부 프레임을 폐기시키는 방법이다. 변경된 CBA[1,2]에서는 CBA 알고리즘을 수행하여 전송 계획을 설정하고 전송하려는 프레임에서 요구되는 전송률이 가용 전송률을 만족시키지 못할 경우에는 해당 프레임을 폐기시킨다. 이 방법은 CBA 알고리즘을 1회 수행하므로 계산 시간이 적게 소요되지만 가용 전송률에 따라 프레임의 폐기 여부가 결정되므로 광범위한 기간에 많은 프레임들이 폐기될 수 있다. 적응형 CBA 방법[1,2]에서는 CBA 알고리즘을 수행하여 전송 계획을 설정하고, 계획된 전송률이 가용 전송률을 만족시키지 못하는 경우에는 가용 전송률을 만족시킬 때까지 런을 구성하는 프레임들중에서 일부 프레임을 폐기시키는데, (그림 5)는 이 방법에서의 재생률 조절 과정이다. (그림 5) (a)는 CBA 알고리즘을 수행하여 생성된 런이며, (그림 5) (b)는 초당 30개의 프레임을 재생해야 하는 경우에 요구되는 전송률이 가용 전송률보다 커서 재생률 조절이 필요한 경우이다. (그림 5) (c)는 재생률을 25 fps로 감소시켰으나 아직도 런의 전송률이 가용 전송률보다 커서 재생률을 (그림 5) (d)에서와 같이 20 fps로 감소하여 가용 전송률을 만족시켜야 한다. 이 방법은 네트워크 트래픽이 변동될 때마다 런을 생성하기 위해

CBA 알고리즘을 수행하므로 계산 시간이 많이 소요된다.



(그림 5) 적응형 CBA 방법에서의 재생률 조절 과정

3. 실험 결과

e-PCRTT를 개선한 알고리즘들의 성능은 두 가지 환경에서 평가된다. 첫째, 네트워크 트래픽을 고려하지 않는 환경에서 개선된 e-PCRTT 알고리즘 1과 개선된 e-PCRTT 알고리즘 2의 성능을 평가하기 위하여 첨두 전송률[3,6,7], 평균 전송률 증가 양을 비교한다. 이때 실험에 사용된 버퍼 크기는 512KB부터 16MB[1,2]까지 다양하게 설정한다.

둘째, 네트워크 트래픽을 고려하는 환경에서 개선된 e-PCRTT 알고리즘 1과 개선된 e-PCRTT 알고리즘 2의 성능을 평가하기 위하여 적응형 대역폭 할당 방법에서의 전송 계획으로 이들을 각각 적용하여 폐기되는 프레임 양을 비교한다. 이때 사용된 버퍼 크기는 1MB[1,2]이고 가용 전송률은 비디오 스트림을 구성하는 프레임들의 평균 바이트 수의 0.4배부터 1.6배까지 다양하게 설정하여 실험한다. 두 경우 모두에서 사용된 비디오 데이터는 E.T.90[10]이다.

<표 2>는 개선된 e-PCRTT 알고리즘 1과 개선된 e-PCRTT 알고리즘 2에서의 첨두 전송률을 비교한 결과이다. 이 값이 클수록 가용 전송률이 낮은 경우에 폐기되는 프레임이 많아질 수 있다. 개선된 e-PCRTT 알고리즘 2에서의 첨두 전송률은 버퍼 크기가 1MB와 4MB, 그리고 8MB인 경우에 개선된 e-PCRTT 알고리즘 1에서보다 작는데, 이는 개선된 e-PCRTT 알고리즘 2에서 전송률 증가가 요구되는 경우에 전송률 증가를 완만하게 변화시키기 때문이다.

<표 2> 첨두 전송률 비교

구간 버퍼 크기	512KB	1MB	2MB	4MB	8MB	16MB
개선된 e-PCRTT 알고리즘 2 [5]	18.3	17.4	15.6	14.2	11.4	10.4
개선된 e-PCRTT 알고리즘 1 [4]	18.3	17.5	15.6	14.4	12.3	10.4

<표 3>은 평균 전송률 증가 양을 비교한 결과이다. 이 값은 전송률 증가가 요구되는 구간들의 전송률 증가량의 합을 전송률 증가 횟수로 나눈 몫이다. 이 값이 클수록 새로운 구간의 전송률이 현재 구간의 전송률에 비해 크게 증가되어 가용 전송률이 낮은 경우에는 새로운 구간에서 폐기되는 프레임 양이 많아질 수 있다.

개선된 e-PCRTT 알고리즘 2에서의 전송률 증가량이 개선된 e-PCRTT 알고리즘 1에서보다 작는데, 이는 전송률 증가 양을 최소로 하는 전송 계획을 세우기 때문이다.

<표 3> 전송률 증가 양 비교

알고리즘 버퍼 크기	512KB	1MB	2MB	4MB	8MB	16MB
개선된 e-PCRTT 알고리즘 2 [5]	0.19	0.23	0.32	0.32	0.55	0.55
개선된 e-PCRTT 알고리즘 1 [4]	0.97	1.17	1.11	0.96	1.33	0.86

<표 4>는 폐기되는 프레임 양을 비교한 결과이다. 개선된 e-PCRTT 알고리즘 2에서의 폐기되는 프레임 양이 개선된 e-PCRTT 알고리즘 1에서보다 적거나 같은데, 이는 전송률 증가가 요구되는 경우에 개선된 e-PCRTT 알고리즘 1에서보다 전송률 증가가 완만하게 변화되어서 프레임을 적게 폐기시켜도 가용 전송률을 만족시킬 수 있기 때문이다. 두 방법 모두에서 가용 전송률이 1.6배인 경우에 폐기되는 프레임 양이 0인데, 이는 가용 전송률이 충분히 커서 모든 구간에서 전송 계획에서 요구되는 전송률이 가용 전송률을 만족시켰기 때문이다.

<표 4> 폐기되는 프레임 양 비교

가용전송률 알고리즘	0.4	0.6	0.8	1.0	1.2	1.4	1.6
개선된 e-PCRTT 알고리즘 2 [5]	118682	78685	40125	14679	2013	198	0
개선된 e-PCRTT 알고리즘 1 [4]	119856	79181	40128	14679	2013	198	0

#### 4. 결론 및 추후 연구 방향

e-PCRTT를 개선한 알고리즘[4]에서는 전송률 증가 양을 고려하지 않고 전송률을 설정하여서 급격하게 높은 전송률이 요구될 수 있다. 이는 네트워크 트래픽이 고려되는 경우에 전송 계획으로 사용하려는 경우 많은 프레임이 폐기될 수 있다. 이를 개선한 알고리즘[5]에서는 전송률 증가가 요구되는 경우에 전송률 증가 양을 최소로 하는 프레임을 검색하여 이 프레임까지를 런으로 설정한다.

본 논문에서는, e-PCRTT를 개선한 알고리즘[4]와 [4]를 다시 개선한 알고리즘[5]의 성능을 평가하기 위하여 첨두 전송률, 평균 전송률 증가 양, 폐기되는 프레임 양을 비교하였다. 그 결과, [4]를 개선한 알고리즘[5]에서의 첨두 전송률, 평균 전송률 증가 양, 폐기되는 프레임 양이 모두 우수하였다.

추후에는 다양한 비디오 소스로써 기존의 여러 스무딩 알고리즘과 제안 알고리즘의 성능을 비교 분석할 예정이다.

#### 참고문헌

- [1] Wu-chi Feng, Ming Liu, "Critical Bandwidth Allocation Technique for Stored Video Delivery Across Best-Effort Network", pp.25, (OSU- CISRC-8/98-TR32) Electronic report under 1998/TR32.ps
- [2] Wu-Chi Feng, Ming Liu, "Extending critical bandwidth allocation Techniques for stored video delivery across best-effort networks", International Journal of COMMUNICATION SYSTEMS Int.J.Commun.sust, 2001.
- [3] W. Feng, S. Sechrest, "Critical Bandwidth Allocation for the Delivery of Compressed Pre-recorded Video," Computer Communications, Vol.18, No.10, pp.709-717, Oct. 1995.
- [4] 이면재, 박도순, 이준용, "가변 비트율 비디오 전송을 위한 효율적인 스무딩 알고리즘", 한국정보처리학회 논문지, 제 11-C권 제 7호, 2004.12.
- [5] 이면재, 박도순, "적용형 대역폭 할당 방법을 위한 효율적인 전송 계획", 한국정보처리학회논문지, 제14-C권, 2007.6
- [6] J. McManus and K.Ross, "Video on demand over ATM:Constant-rate Transmission and Transport",in proc.of ACM SIGMETRICS, pp.222-231, May 1996.
- [7] W.Feng, F.Jahanian, S.Sechrest, "An Optimal Bandwidth Allocation Strategy for the Delivery of Compressed Pre-recorded Video", ACM/Springer-Verlag Multimedia Systems, vol. 5, no. 5, pp.297-309, Sept 1997.
- [8] 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.
- [9] 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.
- [10] <http://www.cis.ohio-state.edu/~wuchi>