

적응형 대역폭 할당 방법을 위한 효율적인 전송 계획

이 면 재[†] · 박 도 순^{**}

요 약

적응형 대역폭 할당 방법에서는 가변 비트율로 저장된 비디오 데이터에 대한 전송 계획을 세우고 네트워크 트래픽을 고려하여 전송하는데, 전송 계획으로 CBA 알고리즘이 사용된다. 그러나, CBA 알고리즘에서는 전송률 증가 구간의 크기가 감소 구간의 크기보다 일반적으로 크며, 전송률이 증가될 때에 오버플로우 경계선에서 전송률이 변화되므로 가용 전송률이 작은 경우에는 폐기되는 프레임의 양이 많아지게 된다. 본 논문에서는 이를 개선하기 위하여 언더플로우 경계선과 오버플로우 경계선의 중간에서 전송률이 변화되지만, 전송률의 증가가 필요한 경우에는 증가 양을 최소화 하는 스무딩 알고리즘을 제안한다. 제안 알고리즘과 CBA 알고리즘을 적응형 대역폭 할당 방법의 전송 계획으로 사용하였을 때, 최소 재생률, 평균 재생률, 재생률 변화량, 그리고 폐기되는 프레임 양을 비교하여 제안 알고리즘의 성능이 우수함을 보였다.

키워드 : 스무딩, 가용 전송률

An Efficient Transmission Plan for Adaptable Bandwidth Allocation Technique

Myoun-Jae Lee[†] · Do-Soon Park^{**}

ABSTRACT

In the adaptable bandwidth allocation technique, a transmission plan for variable rate video data is made by the CBA algorithm and the data is transmitted by considering network traffic. But the CBA algorithm produces a transmission plan where the size of the increasing interval of transmission rate is generally larger than the size of the decreasing interval. And the transmission rate in CBA algorithm is changed in overflow curve during the increasing interval of transmission rate. This may cause many frames to be discarded when available transmission rate is small. In this paper, a smoothing algorithm is proposed, where transmission rate is changed in the middle of underflow curve and overflow curve, but the transmission rate increases at the minimum. In order to show the performance, the proposed algorithm and a CBA algorithm were applied to a transmission plan in the adaptable bandwidth allocation technique, and the minimum frame rates, the average frame rates, the variation of frame rates, and the numbers of discarded frames were compared in both algorithms.

Key Words : Smoothing, Available Network Traffic

1. 서 론

비디오 데이터 전송시에 가변 비트율 방법은 프레임을 구성하는 비트 수의 차이가 심하므로 이것을 그대로 전송하려면 QoS를 보장하기 위해서 전송률을 급격히 증가시켜야 되는 버스트(Burst) 현상이 발생할 수 있다[1-3, 8]. 이 문제를 해결하기 위해 스무딩(Smoothing), 적응형 비디오 전송(Adaptable Video Transmission), 적응형 대역폭 할당(Adaptable Bandwidth Allocation) 방법 등이 사용된다[2-4]. 스무딩은 네트워크의 트래픽을 고려하지 않고 전송 계획을 수립하여 비디오 스트림을 전송하는 방법이다. 이 방법은 첨두 전송률, 전송률 변화량 등의 특정 요소에 최적화 된

전송을 할 수 있지만 서버에서 세운 전송 계획대로 네트워크 자원이 확보되는 환경에서 제한적으로 사용된다. 적응형 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. 적응형 대역폭 할당은 스무딩 알고리즘을 사용하여 전송 계획을 수립하고, 전송 계획에서 요구되는 전송률이 네트워크의 가용 전송률 보다 큰 경우에는 가용 전송률을 만족시킬 때까지 일부 프레임을 폐기하여 재생률을 감소시키는 형태로 네트워크 트래픽을 반영한다. 이 방법에는 변경된 CBA(Modified Critical Bandwidth Allocation)와 적응형 CBA(Adaptable CBA)가 있으며, 네트워크의 트래픽이 변동적인 인터넷과 같은 환경에도 적용될 수 있고 특정 요소를 최적화하면서 전송할 수 있다[2-4].

[†] 정 회 원 : 남서울대학교 멀티미디어학과 교수

^{**} 종 신 회 원 : 홍익대학교 컴퓨터공학과 교수

논문접수 : 2006년 12월 13일, 심사완료 : 2007년 4월 25일

〈표 1〉 CBA 알고리즘에서 전송률 증가 및 감소가 요구되는 구간의 크기 비교

구간	버퍼 크기	64K	128K	256K	512K	1M	2M	4M	8M	16M
감소 구간 비율		52%	47%	46%	43%	43%	47%	50%	45%	42%
증가 구간 비율		48%	53%	54%	57%	57%	53%	50%	55%	58%

그러나 네트워크 트래픽을 반영할 때에 CBA 전송 계획에서는 전송률 증가 구간 즉 오버플로우 경계선에서 전송률이 변화되는 구간의 크기가 전송률 감소 구간 즉 언더플로우 경계선에서 변화되는 구간의 크기보다 일반적으로 크다. <표 1>은 이를 보여주는 것으로 사용된 비디오 데이터는 E.T.90[2-4]이고 버퍼 크기는 64KB부터 16M[2-4]까지이다. 이때 전송 계획에서 요구되는 전송률이 네트워크의 가용 전송률보다 크다면 전송률이 증가되는 경우에 폐기시켜야 되는 프레임의 양이 전송률이 감소되는 구간인 경우에서보다 많아진다. 또한, 네트워크의 트래픽을 반영할 때에 MCBA (Minimum Changes Bandwidth Allocation), MVBA (Minimum Variability Bandwidth Allocation) 알고리즘을 전송 계획으로 사용하는 경우에도 이 알고리즘들은 전송률 증가가 오버플로우 경계선에서 이루어지므로 가용 전송률이 작은 경우에는 폐기되는 프레임이 많아지게 된다. 그러므로 네트워크의 트래픽으로 인해 폐기되는 프레임의 전체 양을 줄이기 위해서는 전송률 증가의 구간을 줄이거나 또는 전송률 증가 양을 줄여야 한다. 이를 위해서는 오버플로우 경계선에서 전송률이 변화되는 상황을 탈피해야 한다.

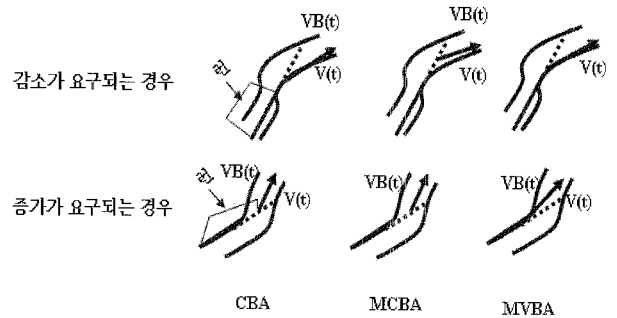
본 논문에서는 네트워크의 트래픽으로 인해 폐기되는 프레임의 전체 양을 줄이기 위해 언더플로우 경계선과 오버플로우 경계선의 중간에서 전송률이 변화[6]되지만, 전송률의 증가가 필요한 경우에는 전송률 증가 양을 줄일 수 있도록 전송률 증가 양의 변화를 최소로 하는 프레임을 검색하여 이를 런(run)으로 설정하는 스무딩 알고리즘을 제안한다. 제안 방법에 의한 전송률 증가량은 CBA 방법보다 작아지지만 전송률이 감소되는 양은 CBA 방법보다 커지게 된다. 그러나 전송률 증가 양을 줄여서 폐기되는 프레임 양의 감소가 전송률 감소로 인해 폐기되는 프레임 양의 증가보다 커지므로 전체적으로 폐기되는 프레임의 양이 감소하게 된다.

논문의 구성은 다음과 같다. 2장에서는 관련 연구를 살펴보고, 3장에서는 제안 알고리즘을 설명한다. 4장에서는 실험 결과를 설명하고, 5장에서는 결론 및 추후 연구 방향을 기술한다.

2. 관련 연구

2.1 스무딩

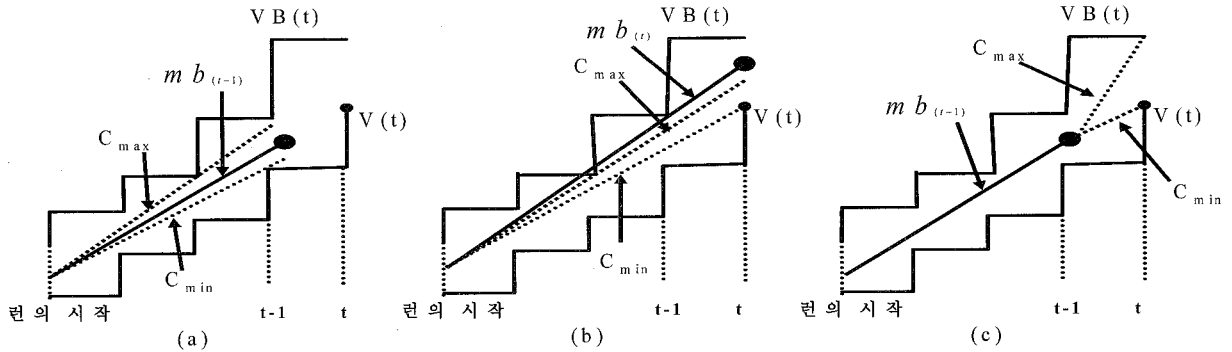
스무딩 방법은 네트워크의 트래픽을 고려하지 않고 비디오 데이터를 동일한 전송률로 전송할 수 있는 전송 계획으로, QoS를 만족하면서 전송률 변화 횟수, 전송률 변화량, 그리고 버퍼 크기와 같은 특정 요소를 최적화하기 위한 목적



(그림 1) CBA, MCBA, MVBA의 전송률 조절 방법

으로 다양한 스무딩 알고리즘들이 연구되어 왔다. (그림 1)은 CBA, MCBA, MVBA 알고리즘의 전송률 조절 방법이다 [7,8]. $V(t)$ 는 언더플로우 경계선으로 비디오 데이터를 구성하는 첫 번째 프레임부터 프레임 t 까지의 누적된 바이트 수이며 $VB(t)$ 는 오버플로우 경계선으로 언더플로우 경계선에 클라이언트의 버퍼 크기를 더한 값이다. 연장 구간인 점선은 현재의 전송률에 의해 언더플로우 경계선(오버플로우 경계선)을 만나는 경우에 이 지점부터 오버플로우(언더플로우)를 발생시키는 프레임까지의 구간을 의미하며, 이 연장 구간에 속한 프레임들 중에서 동일한 전송률로 전송할 수 있는 연속적인 프레임들인 새로운 런의 시작 프레임을 선택하는 방법에 따라 스무딩 알고리즘의 목적이 달라진다.

CBA 알고리즘[5]에서는 현재 전송률에 의해 오버플로우가 발생된다면 연장 구간의 첫 프레임을 새로운 런의 시작 프레임으로 설정하며, 현재 전송률에 의해 언더플로우가 발생하는 경우에는 연장 구간에 속한 프레임들 중에서 새로운 전송률로 언더플로우나 오버플로우가 발생되지 않고 가장 많은 프레임을 전송할 수 있는 프레임을 검색하여 이 프레임을 새로운 런의 시작 프레임으로 설정하므로써 전송률 증가 횟수를 최소화한다. MCBA 알고리즘[10]의 목적은 전송률 변화 횟수의 최소화인데, 이를 위해 다음 런의 전송률을 감소 또는 증가시켜야 할 때에 연장 구간에 있는 프레임 중에서 새로운 전송률에 의해 가장 많은 프레임을 전송할 수 있는 프레임을 검색하여 이 프레임을 다음 런의 시작 프레임으로 설정한다. MVBA 알고리즘[11,15]은 전송률 변화량을 최소화하는 것이 목적이며, 이를 위해 다음 런의 전송률을 증가 또는 감소시켜야 할 때에 연장 구간의 첫 프레임을 다음 런의 시작 프레임으로 설정한다. 구현의 용이함과 네트워크의 전송률 예약에 대한 통신 비용을 줄이려는 목적을 갖는 CRTT(Constant Rate Transmission and Transport) 알고리즘[9]에서는 런의 개수가 1개이어서 클라이언트 버퍼 크기와 초기 지연 시간이 크고, 침투 전송률의 이용률이 낮을 수 있다[9]. 이러한 단점을 개선하여 PCRTT(Piecewise Constant Rate Transmission and Transport) 방법[7-9]에서는 전송률 변화 횟수 즉 구간의 개수를 설정하므로써 CRTT 방법보다 버퍼 크기를 줄였지만 언더플로우를 피하기 위해 모든 구간에 동일한 옵션 값을 적용하므로 버퍼 크기가 아직도 커질 수 있다[7,8]. e-PCRTT(enhanced-PCRTT)



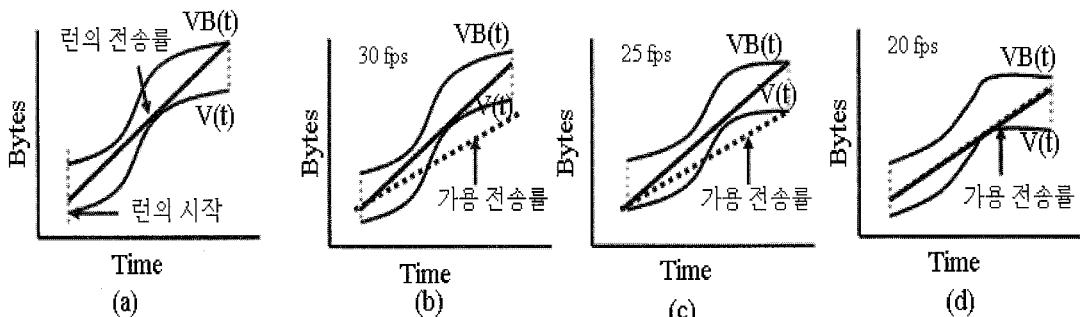
(그림 2) 개선된 e-PCRTT 알고리즘[6]에서의 전송률 조절 방법

알고리즘[13]에서는 버퍼 크기가 커질 수 있는 PCRTT 방법의 단점을 개선하기 위하여 초기 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[6]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 t를 검색해야 하는 상황이다. (그림 2) (b)는 프레임 t에서 설정하려는 전송률 $mb_{(t)}$ 가 QoS를 만족하지 않아서 프레임 t는 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.

2.2 적응형 대역폭 할당

적응형 대역폭 할당은 비디오 데이터에 대한 전송 계획을 세우고 네트워크 트래픽을 고려하여 전송하는 방법이다. 즉,

스무딩 알고리즘에 의한 전송 계획에 의해 동일한 전송률로 전송할 수 있는 프레임들의 그룹을 설정하여, 이 그룹의 전송률이 네트워크의 가용 전송률보다 큰 경우에는 가용 전송률을 만족시킬 때까지 그룹을 구성하는 프레임들 중에서 일부 프레임을 폐기시키는 방법이다. 이 방법에는 변경된 CBA와 적응형 CBA가 있는데, 변경된 CBA에서는 CBA 알고리즘을 수행하여 전송 계획을 세우고 전송하려는 프레임에서 요구되는 전송률이 가용 전송률을 만족시키는 경우에는 해당 프레임을 계획대로 보내지만 그렇지 않을 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[4]. 적응형 CBA 방법에서는 1초마다 CBA 알고리즘을 수행하여 런을 구하고, 이 런의 전송률이 가용 전송률을 만족시키지 못하는 경우에는 가용 전송률을 만족시킬 때까지 런을 구성하는 프레임들중에서 일부 프레임을 폐기시키는데, (그림 3)은 이 방법에서의 재생률 조절 과정이다[2,3]. (그림 3) (a)는 CBA 알고리즘을 수행하여 생성된 런이며, (그림 3) (b)는 30 fps, 즉 초당 30개의 프레임을 재생해야 하는 경우에 런의 전송률이 가용 전송률보다 커서 재생률 조절이 필요한 경우이다. (그림 3) (c)는 재생률을 25로 감소시켰으나 아직도 런의 전송률이 가용 전송률보다 커서 재생률을 (그림 3) (d)에서와 같이 20 fps로 감소하여 가용 전송률을 만족시켜야 한다. 이 방법은 다른 스무딩 알고리즘에 비해 CBA 알고리즘 자체의 계산 시간이 클 뿐만 아니라, 1초마다 런을 생성하기 위해 CBA 알고리즘을 수행하므로 계산 시간이 많이 소요된다[2-4].



(그림 3) 적응형 CBA 방법에서의 재생률 조절 과정

3. 제안 알고리즘

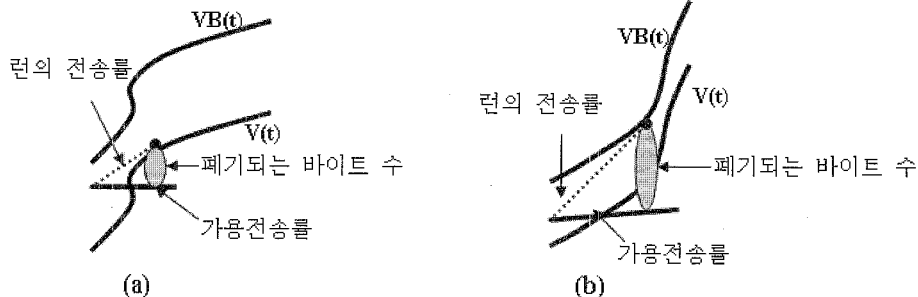
3.1 배경

CBA, MCBA, MVBA 스무딩 알고리즘들에서는 (그림 1)에서와 같이 언더플로우 경계선 또는 오버플로우 경계선을 중심으로 전송률이 변화된다. 이 알고리즘들이 적응형 대역폭 할당 방법에서의 전송 계획으로 사용될때, 만일 이 전송률이 네트워크의 가용 전송률보다 크다면 가용 전송률을 만족시키기 위해 프레임들이 폐기되어야 한다. (그림 4)는 이러한 상황인데, (그림 4) (a)는 전송률의 감소가 요구되어 런의 전송률이 언더플로우 경계선과 만나는 경우이고, (그림 4) (b)는 전송률의 증가가 요구되어 오버플로우 경계선과 만나는 경우이다. 두가지 경우에 가용 전송률이 같다면, 런의 전송률은 (그림 4) (b)가 (그림 4) (a)보다 크기 때문에 (그림 4) (a)에서 보다 (그림 4) (b)에서 폐기되는 프레임 양이 훨씬 많게 된다. 특히 CBA 알고리즘을 사용하는 경우에는 <표 1>에서와 같이 전송률 증가 구간이 감소 구간보다 일반적으로 크기 때문에 폐기되는 프레임의 양이 많아질 수 있다. 따라서 폐기되는 프레임의 양을 줄이기 위해서는 전송률 증가량을 줄이든지 또는 증가 구간의 크기를 줄여야 한다. 개선된 e-PCRTT 알고리즘[6]에서는 (그림 2)의 과정으로 현재 런을 구하는데, 전송률 증가량을 고려하지 않고 새로

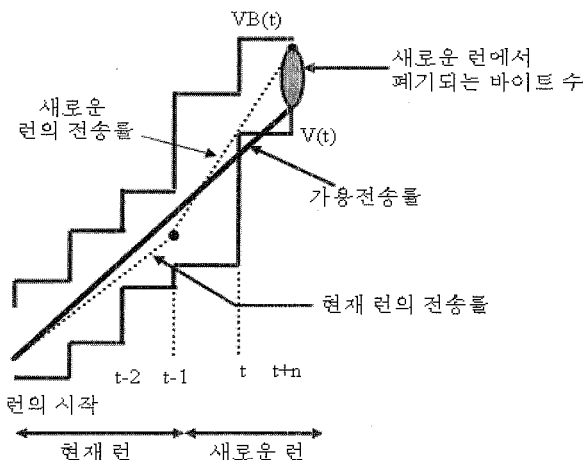
운 런을 구하므로 새로운 구간에서 급격하게 높은 전송률이 요구될 수 있다. 그래서, 이 알고리즘을 적응형 대역폭 방법에서의 전송 계획으로 적용하는 경우에도 앞에서 언급한 것과 유사한 상황이 발생할 수 있다. (그림 5)는 이러한 상황의 예인데, 현재 런에서는 가용 전송률이 런의 전송률보다 커서 폐기되는 프레임이 없지만 새로운 런에서는 가용 전송률이 런의 전송률보다 작아서 프레임들이 폐기되어야 한다. 그러나 (그림 6)에서와 같이 현재 런을 구성하는 프레임들 중에서 프레임 (t-2)까지를 현재 런으로 완성하여 현재 런과 새로운 런의 전송률 증가를 완만하게 변화시킨다면, 가용 전송률을 만족시키지 위해 폐기시켜야 되는 프레임 양이 (그림 5)에서 보다 적어질 수 있다.

3.2 제안 알고리즘

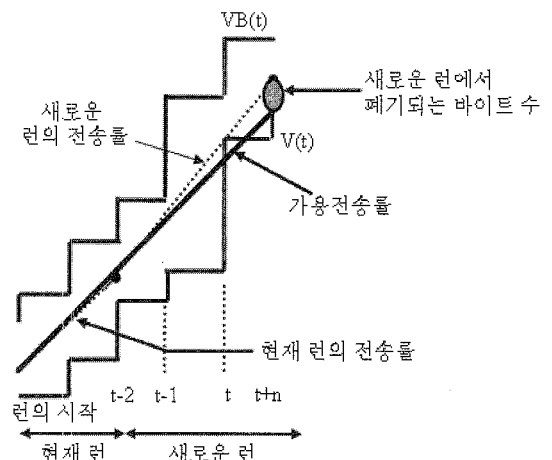
앞절에서 언급한 바와 같이 폐기되는 프레임의 양을 줄이기 위해, 제안 방법에서는 언더플로우 경계선과 언더플로우 경계선의 중간에서 전송률이 변화[6]되지만, 전송률의 증가가 요구되는 경우에는 현재 런을 구성할 수 있는 프레임들 중에서 현재 런과 새로운 런의 전송률이 가장 완만하게 변화될 수 있는 프레임들 검색하여 이 프레임들을 새로운 런의 시작 프레임으로 설정하도록 한다. (그림 7)은 제안 방법에 의한 전송률 조절 방법이다. (그림 2)에서와 같이 현재 런과



(그림 4) 네트워크 트래픽이 고려되는 경우에 CBA, MCBA, MVBA 알고리즘들에서 폐기되는 프레임 양의 비교



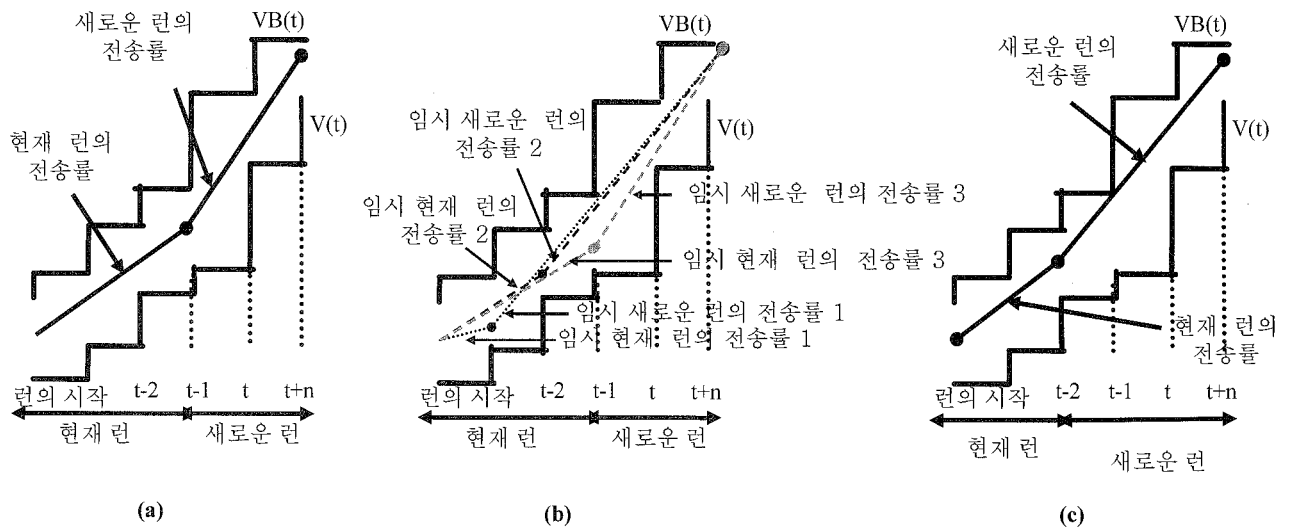
(그림 5) 개선된 e-PCRTT 알고리즘[6]에서의 폐기되는 프레임 양의 예



(그림 6) 폐기되는 프레임 양을 줄이기 위한 방법의 예

새로운 런을 구하고, 현재 런의 전송률로 프레임들을 전송하는 경우에 언더플로우는 발생되는지 또는 오버플로우는 발생되는지 여부를 점검한다. (그림 7) (a)는 임시 현재 런의 전송률로 전송하는 경우에 언더플로우가 발생되어 새로운 런의 전송률이 증가되어야 하는 경우인데, 이 경우에 (그림 7) (b)에서와 같이 현재 런의 시작 프레임부터 프레임 (t-1)까지에 속한 프레임들에 대해 각 프레임에서의 전송률을 임시 현재 런의 전송률로 다시 설정하고 이 프레임을 시작점으로 하여 임시 새로운 런의 전송률을 다시 계산하여 전송률의 증가가 가장 완만하게 되는 프레임을 검색한다. 그래서, (그림 7) (c)에서와 같이 검색된 프레임 (t-2)까지를 현재 런으로 결정하고 프레임 (t-2)부터 프레임 (t+n)까지를 새로운 런으로 설정한다. 새로운 런의 전송률 감소가 요구되는 경우에는 (그림 2)에서와 동일한 방법으로 런과 이 런의 전송률을 설정한다. 이와 같은 원리를 사용한 제안 알고리즘이 <표 2>인데, n은 비디오 스트림을 구성하는 프레임

개수이며 t_s 는 런의 시작 프레임이다. 단계 (6)은 런의 시작 프레임부터 현재 검색중인 프레임 t까지의 전송률인 $mb(t)$ 를 계산하는 과정으로 q는 버퍼에 채워져 있는 바이트 수이다. 단계 (7)에서 (14)까지는 현재 런의 설정과 이 런의 전송률을 설정하는 과정이다. 단계 (7)에서는 프레임 t까지의 전송률 $mb(t)$ 가 C_{max} 보다 크거나 C_{min} 보다 작은지, 즉 QoS를 보장하는 범위를 벗어나는지를 검색한다. 단계 (8)의 next_run_flag는 새로운 런에서 전송률의 증가 또는 감소가 요구되는지를 나타내는 플래그인데, 이 값이 증가인 경우에는 단계 (10)부터 (11)까지에서와 같이 임시 현재 런과 임시 새로운 런을 구하여 임시 현재 런에 속한 프레임들 중에서 임시 현재 런의 전송률과 임시 새로운 런의 전송률이 가장 완만하게 증가되는 프레임을 검색하여 이 프레임을 새로운 런의 시작 프레임으로 설정한다. 단계 (13)의 output($t_s \sim t-1, mb_{(t-1)}$)는 프레임 t_s 부터 프레임 (t-1)까지의 전송률을 $mb_{(t-1)}$ 로 설정하는 함수이다.



(그림 7) 제안 방법에 의한 전송률 조절 방법

<표 2> 제안 알고리즘

```

proposed algorithm()
{
(1) t=ts+1
(2) compute Cmax, Cmin
(3) n=number of last frame
(4) Repeat
(5) t=t+1
(6) mb(t)=((VB[t]-V[t])/2.0 - (V[ts]+q) )/(t-ts)
(7) if ( (Cmax < mb(t)) or (Cmin > mb(t)) )
(8) determine next_run_flag
(9) if (next_run_flag==Inc)
(10) determine temporal current run and temporal next run
(11) t=find the frame with the smallest rate increase between temporal current run and temporal next run
(12) endif
(13) output(ts~t-1,mb(t-1))
(14) endif
(15) compute Cmin, Cmax, q
(16) UNTIL t==n
}
    
```

<표 3> 시간 복잡도 비교

알고리즘	시간 복잡도
MVBA[11,15]	$O(n^2)[11] \sim O(n)[15]$
CBA[5]	$O(n^2 \log n)[5] \sim O(n)[17]$
MCBA[16,17]	$O(n^2 \log n)[16] \sim O(n)[17]$
e-PCRTT[13]	$O(n)$
개선된 e-PCRTT[6]	$O(n)$
제안 알고리즘	$O(n^2 \log n) \sim O(n)$

<표 3>은 제안 알고리즘과 기존 스무딩 알고리즘들의 시간 복잡도 비교이다. 제안 알고리즘의 시간 복잡도는 전송률의 증가가 요구되는 경우에 전송률 증가량이 최소가 되는 프레임을 선택하는 과정이 필요하여서 CBA, MCBA, MVBA 알고리즘과 동일하고, 개선된 e-PCRTT 알고리즘[6]보다 크다.

4. 실험 결과

제안 알고리즘의 성능을 평가하기 위해 적용형 대역폭 할당 방법에서의 전송 계획으로 제안 알고리즘과 CBA 알고리즘을 각각 적용하여 평가 요소인 최소 재생률[2-4], 평균 재생률[2-4], 재생률 변화량[2-4], 그리고 폐기되는 프레임 양을 비교한다. 실험에 사용된 비디오 데이터는 E.T.90[2,3,14]이며 버퍼 크기는 1M로 고정하고 가용 전송률은 비디오 스트림을 구성하는 프레임들의 평균 바이트 수의 0.4배부터 1.6배[2-4]까지 다양하게 설정한다.

<표 4>는 각 가용 전송률에 대해 최소 재생률을 비교한 결과이다. 최소 재생률은 비디오 스트림을 재생하는 모든 구간의 재생률들 중에서 가장 낮은 재생률인데, 이 값이 클수록 클라이언트에서는 좋은 화질의 비디오를 시청할 수 있다. 가용 전송률이 0.8과 1, 그리고 1.2인 경우에 제안 방법에서의 최소 재생률이 CBA 알고리즘에서 보다 크고, 이를 제외하면 동일한 결과를 보인다. 이는 제안 알고리즘이 CBA 알고리즘에서보다 프레임들의 바이트 수가 큰 구간의 전송률을 완만하게 설정하여서 가용 전송률을 만족시키기 위하여 폐기되는 프레임이 적었기 때문이다.

<표 5>는 각 가용 전송률에 대해 평균 재생률을 비교한 결과이다. 이 값이 클수록 클라이언트에서는 평균적으로 좋은 화질을 얻을 수 있다. 두 알고리즘의 평균 재생률이 모든 가용 전송률에서 같은데, 이는 두 알고리즘들이 비디오 스트림을 구성하는 모든 프레임들에 대한 바이트 수를 바탕으로 전송 계획을 세우기 때문이다. 가용 전송률이 1.6인 경우에 평균 전송률이 30인데, 이는 가용 전송률이 충분히 커서 거의 모든 구간의 재생률이 30으로 설정되기 때문이다. 두 알고리즘 모두에서 가용 전송률이 클수록 평균 재생률이 커지는데, 이는 프레임을 적게 폐기시켜도 런의 전송률이 가용 전송률을 만족시킬 수 있기 때문이다.

<표 6>은 각 가용 전송률에 대해 재생률 변화량을 비교한 결과로써, 이 값이 적을수록 클라이언트에서는 화질의 변화를 느낄 수 없다. 제안 알고리즘의 재생률 변화량은 가용

<표 4> 최소 재생률 비교

가용 전송률 \ 알고리즘	0.4	0.6	0.8	1	1.2	1.4	1.6
제안 알고리즘	7	10	14	17	19	22	25
CBA	7	10	13	16	18	22	25

<표 5> 평균 재생률 비교

가용 전송률 \ 알고리즘	0.4	0.6	0.8	1	1.2	1.4	1.6
제안 알고리즘	12.1	18.4	24.1	27.7	29.5	29.9	30
CBA	12.1	18.4	24.1	27.7	29.5	29.9	30

<표 6> 재생률 변화량 비교

가용 전송률 \ 알고리즘	0.4	0.6	0.8	1	1.2	1.4	1.6
제안 알고리즘	3.6	4.6	4.1	2.9	2.3	2.6	2.1
CBA	3.6	4.3	4.2	3.2	2.8	2.8	2.2

<표 7> 제안 알고리즘에서 전송률 증가 및 감소 구간의 크기 비교

구간 크기 \ 버퍼 크기	64K	128K	256K	512K	1M	2M	4M	8M	16M
감소 구간 비율	60%	57%	57%	56%	56%	67%	50%	58%	62%
증가 구간 비율	40%	43%	43%	44%	44%	33%	50%	42%	38%

<표 8> 폐기되는 프레임 양 비교

가용 전송률 \ 알고리즘	0.4	0.6	0.8	1	1.2	1.4	1.6
제안 알고리즘	109305	71598	36706	14564	3433	505	179
CBA	112834	73741	37873	14753	3524	509	186

전송률이 0.6인 경우에 CBA 알고리즘보다 큰데, 이는 제안 알고리즘의 전송률 감소량이 CBA 알고리즘에 비해 큰 경우가 일부 발생되어서 폐기되는 프레임 양이 크게 변화되기 때문이다. 이를 제외하면 제안 알고리즘의 재생률 변화량이 CBA 알고리즘에서보다 낮다. 이는 제안 알고리즘의 전송률 변화량이 CBA 알고리즘에서보다 대부분의 경우에 작아서 재생률 변화량도 감소되기 때문이다.

<표 7>은 <표 1>에서와 같은 상황에서 제안 방법에서의 전송률 증가 구간과 전송률 감소 구간의 크기를 비교한 결과이다. 버퍼 크기가 4M인 경우를 제외하면 전송률의 증가가 요구되는 구간의 크기가 전송률의 감소가 요구되는 구간의 크기보다 크다.

<표 8>은 각 가용 전송률에 대해 폐기되는 프레임 양을 비교한 결과이다. 이 값이 적을수록 동일한 가용 전송률에서 보다 많은 프레임을 재생할 수 있기 때문에 클라이언트 측에서는 보다 좋은 화질을 얻을 수 있다. 제안 알고리즘의

폐기되는 프레임의 양이 CBA 알고리즘에서 보다 모두 적은데, 이는 <표 1>과 <표 7>에서 보는 것처럼 전송률 증가가 요구되는 구간의 크기가 감소되었으며 이때 폐기되는 프레임의 감소 양이 전송률 감소가 요구되는 구간에서의 폐기되는 프레임의 증가 양보다 많기 때문이다. 가용 전송률이 커질수록 폐기되는 프레임 개수는 적어지는데, 이는 가용 전송률이 클수록 구간의 전송률을 만족시킬 가능성이 커지기 때문이다.

5. 결론 및 추후 연구 방향

CBA, MCBA, MVBA 스무딩 알고리즘들을 적응형 대역폭 할당 방법에서의 전송 계획으로 적용하는 경우에는 이 알고리즘들의 전송률 변화중에서 오버플로우 경계선에서 변화되는 경우에는 많은 프레임들이 폐기될 수 있다. 한편 e-PCRTT 알고리즘을 개선한 알고리즘[6]은 새로운 구간의 전송률 증가량을 고려하지 않아서 급격하게 높은 전송률이 요구될 수 있어서, 이 알고리즘을 적응형 대역폭 할당 방법에서의 전송 계획으로 사용하는 경우에도 폐기되는 프레임 양이 많아질 수 있다.

본 논문에서는 폐기되는 프레임의 양을 줄이기 위해 언더플로우 경계선과 오버플로우 경계선의 중간에서 전송률이 변화되지만 전송률의 증가가 요구되는 경우에는 전송률 증가량을 최소로 하는 프레임을 검색하여 전송률 증가량을 줄이는 스무딩 알고리즘을 제안하고 성능을 평가하였다. 그 결과, 제안 알고리즘은 CBA 방법보다 폐기되는 프레임 양이 적어지며, 또한 최소 재생률이 커지고, 재생률 변화량도 대부분의 경우에 적어진다. 그리고, 시간 복잡도는 CBA, MCBA, MVBA 알고리즘과 동일하다.

그러나, 제안 알고리즘에서는 확보된 네트워크 트래픽은 프레임을 전송하는 경우에 변동되지 않아야 하는 제약 사항을 갖고 있다. 추후에는 다른 스무딩 알고리즘들을 전송 계획으로 사용하면서, 다양한 버퍼 크기와 가용 전송률, 그리고 다양한 비디오 소스를 사용하여 제안 알고리즘의 성능을 평가하고 개선할 예정이다.

참고 문헌

- [1] D. Le Gall, "MPEG: A video compression standard for multimedia applications", Communications of the ACM, vol.34, pp.47-58, Apr., 1991.
- [2] 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
- [3] 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.
- [4] Wu-Chi Feng, Ming Lue, Brijesh Krishnaswami, "A Priority Based technique for the Best-Effort Delivery of Stored Video", SPIE/IS&T Multimedia Computing and Networking 1999 Sanjose, California, Jan., 1999.
- [5] 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.
- [6] 이면재, 박도순, 이준용, "가변 비트율 비디오 전송을 위한 효율적인 스무딩 알고리즘", 한국정보처리학회, 제 11-C권 제 7호, Dec., 2004.
- [7] W. Feng and J. Rexford, "Performance evaluation of smoothing algorithms for transmitting pre-recorded VBR video", IEEE Trans. on Multimedia, Sep., 1999.
- [8] Wu-chi Feng, "A Comparison of Bandwidth Smoothing Techniques for the Transmission of Pre-recorded Compressed Video", in Proc. IEEE INFOCOM, pp.58-66, Apr., 1997.
- [9] 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.
- [10] 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, Sep., 1997.
- [11] 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.
- [12] O.Hadar S.Greenberg, "Statistical multiplexing and admission control policy for smoothed video streams using e-PCRTT algorithm", International Conference on Information Technology: Coding and Computing, Mar., 2000.
- [13] 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.
- [14] <http://www.cis.ohio-state.edu/~wuchi>
- [15] J. D. Salehi, "Scheduling Network Processing on Multimedia and Multiprocessor Servers", Ph.D. dissertation, Univ. Massachusetts, Amherst, Sep., 1996.
- [16] W. Feng, et.al., "Optimal Buffering for the Delivery of Compressed Pre-recorded Video", Proc. of the IASTED/ISMM Intl Conference on Networks, Jan., 1995.
- [17] J. Zhang and J. Hui, "Applying traffic smoothing techniques for quality of service control in VBR video transmissions", Computer Communications, pp.375-389, Apr., 1998.



이 면 재

e-mail : mjlee@nsu.ac.kr

1992년 홍익대학교 전자계산학과(학사)

1994년 홍익대학교 전자계산학과(석사)

2006년 홍익대학교 전자계산학과(박사)

2006~현재 남서울대학교 멀티미디어학과
교수

관심분야: 게임 제작, 멀티미디어 통신, 멀티미디어 데이터 방송



박 도 순

e-mail : dspark@hongik.ac.kr

1978년 서울대학교 전자공학(학사)

1980년 한국과학기술원 전산학과(석사)

1988년 고려대학교 수학과(박사)

1980년~1983년 국방과학연구소 연구원

1983년~현재 홍익대학교 컴퓨터공학과 교수

관심분야: 컴퓨터 구조, 설계 자동화