효율적인 자원 이용을 위한 EBA 스무딩 알고리즘의 개선

이면재[0] 박도순
홍익대학교 컴퓨터 공학과
{leemj[0], dspark}@cs.hongik.ac.kr

Improvement on EBA Smoothing Algorithm for Efficient Resource Utilization

Myoun-Jae Lee[0] Do-Soon Park
Dept. of Computer Engineering, Hongik University

요 약

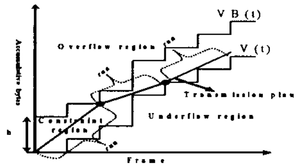
스무딩은 가변 비트율로 저장된 비디오 데이터를 클라이언트로 전송할 때 일련의 고정 비트율로 전송할 수 있도록 전송 계획을 세우는 것이다. 이러한 스무딩 알고리즘들 중에서 e-PCRTT 알고리즘은 전송률 변화 횟수가 주어지고 구간의 크기가 고정이어서 전송률 변화 횟수, 첨두 전송률, 버퍼 이용률 등의 평가 요소들이 증가될 수 있다. 이러한 문제점을 개선하기 위해 전송률 변화 횟수의 제한이 없고 구간의 크기가 가변적인 EBA(Enhanced Bandwidth Allocation)[9,10] 스무딩 알고리즘을 제안한 바 있다. 그러나 이 방법에서는 전송률의 증가 또는 감소를 고려하지 않고 전송률을 변화시키기 때문에 이전 구간의 전송률에 비해 급격하게 높은 전송률이 요구될 수 있으며 증가되는 전송률로 보내야 하는 프레임 개수가 많아질 수 있다. 이는 네트워크 자원의 효율적인 사용을 어렵게 할 수 있다. 따라서, 본 논문에서는 EBA 알고리즘을 개선하여 첨두 전송률과 증가되는 전송률로 보내야 되는 프레임 개수를 감소시키는 알고리즘을 제안한다. 제안된 알고리즘의 성능은 E.T 90의 비디오 소스를 가지고 EBA 알고리즘과 전송률 변화 횟수, 첨두 전송률, 증가되는 전송률로 보내야 하는 프레임 개수 등을 비교 분석하여 평가하였다.

1. 서 론

비디오 데이터는 저장 공간과 효율적인 네트워크 전송을 위하여 압축되어 저장된다. 이러한 압축 방식에는 프레임을 구성하는 비트 수가 동일한 가변 비트율(Variable Bit Rate) 방법과 동일하지 않는 고정 비트율(Constant Bit Rate) 방법이 있다[1]. 특히, 가변 비트율 방법으로 저장된 비디오 데이터는 프레임을 구성하는 비트 수의 차이가 심하므로 이것을 그대로 전송한다면 QoS(Quality of Service)를 보장하기 위해서 전송률을 급격히 증가시켜야 되는 버스트(Burst) 현상이 발생될 수 있다[1,2,3]. 스무딩 기법은 이러한 버스트 현상을 막기 위해 가변 비트율 방법으로 저장된 비디오 데이터를 일련의 고정 비트율로 전송할 수 있도록 비디오 데이터에 대한 전송 계획을 세우는 것이다[1,2,3]. 이러한 스무딩을 위한 알고리즘에는 CBA(Critical Bandwidth Allocation)[4], MCBA(Minimum Changes Bandwidth Allocation)[4], MVBA(Minimum Variability Bandwidth Allocation)[5], PCRTT (Piecewise Constant Rate Transmission and Transport)[6,7], e-PCRTT(enhanced-PCRTT) 알고리즘[8] 등이 있다. 특히, 버퍼 크기가 커질 수 있는 PCRTT 알고리즘의 문제점을 개선한 e-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 평가 요소들이 증가될 수 있다. 이러한 e-PCRTT 알고리즘의 문제점을 개선하기 위하여 전송률 변화 횟수의 제한이 없고 구간의 크기가 가변적인 EBA(Enhanced Bandwidth Allocation)[9,10] 스무딩 알고리즘을 제안한 바 있다[9,10]. 그러나, 이 방법에서는 전송률이 변화될 때 새로운 구간의 전송률의 감소 또는 증가를 고려하지 않고 전송률을 변화시키기 때문에 급격하게 높은 전송률이 요구될 수 있고 증가되는 전송률로 보내야 하는 프레임 개수가 많아질 수 있다. 이는 네트워크 자원 사용의 효율성을 감소시킬 수 있다. 따라서, 본 논문에서는 EBA 알고리즘의 문제점을 개선하기 위하여 전송률의 증가가 필요한 경우에는 전송률 증가량을 최소로 하는 프레임을 검색하여 이 프레임을 동일한 전송률로 전송할 수 있는 구간의 마지막 프레임으로 설정함으로써 첨두 전송률과 증가되는 전송률로 보내야 하는 프레임 개수를 감소시키는 알고리즘을 제안한다. 제안 알고리즘과 EBA 알고리즘의 성능 평가를 위해 E.T 90의 비디오 데이터를 사용하여 전송률 변화 횟수, 첨두 전송률, 버퍼 이용률, 증가되는 전송률로 보내야 하는 프레임 개수를 비교하였다. 본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 기술하고, 3장에서는 제안 알고리즘을 기술한다. 4장에서는 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[7].



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

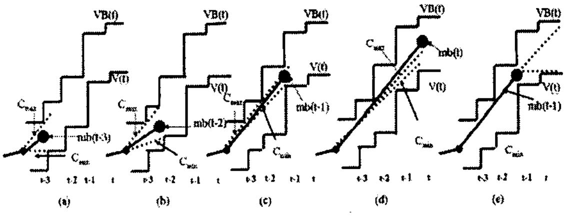
$$VB(t) = b + \sum_{i=0}^{t} f_i \quad \text{...(2)}$$

[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_i는 프레임 i 의 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 발

생되어 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.

이러한 원리를 바탕으로 전송률 변화 횟수, 첨두 전송률, 버퍼 이용률 등의 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. 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 프레임의 언더플로우 경계점과 구간의 끝 프레임의 언더플로우 경계점을 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. 그러나, e-PCRTT 알고리즘에서는 버퍼 크기가 작은 경우에 QoS를 만족시키지 못하는 구간이 발생될 수 있으며 강제적으로 구간마다 전송률을 변화 시켜야 되므로 불필요하게 전송률이 변화 될 수 있다. 이러한 e-PCRTT 알고리즘의 문제점을 개선하기 위하여 전송률 변화 횟수의 제한이 없고 구간의 크기가 가변적인 EBA 알고리즘을 제안한 바 있다[9,10].



[그림 2] EBA 알고리즘의 전송률 조절 방법

[그림 2]는 e-PCRTT 알고리즘의 문제점을 개선한 EBA 알고리즘의 전송률 조절 방법이다[9,10]. Cmin는 런의 시작 프레임부터 현재 검색하고 있는 프레임까지 QoS를 보장하는 전송률 중에서 최소 전송률이고 Cmax은 최대 전송률이다. mb(t-1)는 현재 검색중인 프레임 t-1에 의해 생성되는 전송률로서 런의 시작 프레임의 언더플로우 경계점과 오버플로우 경계점의 평균을 시작점으로 설정하고 프레임 t-1의 언더플로우 경계점과 오버플로우 경계점의 평균을 종점으로 하는 연결선이다. [그림 2] (a)부터 (c)까지에서는 각 프레임에서 설정하려는 전송률이 QoS를 만족시킬 수 있으므로 런의 시작 프레임부터 프레임 (t-1)까지의 프레임들은 동일한 런에 포함시킬 수 있으며, 다음 프레임 t를 동일한 런에 포함시킬 수 있는지를 검색한다. mb(t)가 [그림 2] (d)에서와 같이 Cmin과 Cmax 범위를 벗어나 QoS를 보장할 수 없다면, 현재 런을 완성하고 새로운 런을 구성하여 QoS를 보장할 수 있도록 해야 한다. 즉, [그림 2] (c)와 같이 런의 시작 프레임부터 프레임 (t-1)까지의 전송률을 mb(t-1)로 설정하여 현재 런을 완성하고, 프레임 (t-1)을 새로운 런의 시작 프레임으로 설정한다.

3. 제안 알고리즘

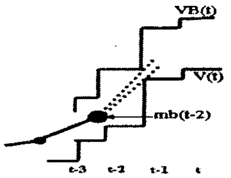
```
proposed algorithm()
{
(1)   t=ts+1
(2)   compute Cmax, Cmin
(3)   n=number of last frame
(4)   Repeat
(5)     t=t+1
(6)    mb(t)=((VB[t]-Ver[t])/2.0 -(V[ts]+q) )/(t-ts)
(7)     if (( ( Cmax < mb(t) ) or (Cmin > mb(t)))
(8)       if (cur_run_flag==Inc)  t= min_frame
(9)       output(ts~t-1,mb(t-1))
(10)       determine cur_run_flag
(11)     endif
(12)     compute Cmin,Cmax,q
(14)     if (cur_run_flag==Dec) continue;
(15)    determine next_run_flag when sending mb(t) rate
(16)     if (next_run_flag==Inc) continue
(17)     compute min_frame
(18)  UNTIL t==n
}
```

[표 1] 제안 알고리즘

EBA 알고리즘에서는 현재 검색되는 런의 전송률의 증가 또는 감소가 요구되는 경우를 고려하지 않고 전송률을 설정하므로 급격하게 높은 전송률이 요구될 수 있으며 증가되는 전송률로 보내야 하는 프레임 개수가 커질 수 있다. 이는 네트워크 자원의 효율적인 사용을 어렵게 할 수 있다.

따라서, EBA 알고리즘을 개선하여 완만하게 전송률이 증가되고 증가되는 전송률로 보내야 하는 프레임의 개수를 줄이는 스무딩 알고리즘을 제안한다. 제안 방법에서는 새로운 런의 전송률의 증가가 필요한 경우에 EBA 알고리즘에서와 같이 새로운 프레임을 검색하는데, 검색되는 프레임들 중에서 이전 전송률에 비해 전송률의 증가가 가장 적은 프레임까지를 새로운 런으로 설정한다.

이러한 원리를 사용한 알고리즘이 [표 1]인데, n은 비디오 스트림을 구성하는 프레임 개수이다. 단계(6)은 런의 시작 프레임부터 현재 검색중인 프레임 t까지의 전송률인 mb(t)를 계산하는 과정으로 q는 버퍼에 채워져 있는 바이트 수이고, ts는 런의 시작 프레임, cur_run_flag는 현재 검색되는 런의 전송률이 증가 또는 감소가 요구되는지를 나타내는 플래그이다. 단계 (7)에서 (11)까지는 프레임 t까지의 전송률 mb(t)가 Cmax보다 크거나 Cmin 보다 작게 된다면, 즉 QoS를 보장하는 범위를 벗어난다면 런을 완성하는 과정이다. 이때 런의 전송률이 감소되어야 하는 경우에는 [그림 2]의 EBA 방법과 동일하게 런을 완성하고, 현재 런의 전송률이 증가되어야 하는 경우에는 이전 런의 전송률에 비해 가장 완만하게 증가되는 전송률을 갖는 프레임인 min_frame까지를 현재 런으로 설정하고 mb(min_frame)를 런의 전송률로 설정한다. output (ts~t-1, mb(t-1))는 프레임 ts부터 프레임 (t-1)까지의 전송률을 mb(t-1)로 설정하는 함수이다. 단계 (14)에서 (17)까지는 이전 런의 전송률에 비해 현재까지 검색되는 프레임들 중에서 가장 완만하게 전송률을 증가시킬 수 있는 전송률을 갖는 프레임, 즉 min_frame을 설정하는 과정이다. 제안 알고리즘에 의한 [그림 2]의 전송률 조절 예는 [그림 3]이다. 전송률 증가가 가장 적은 전송률을 갖는 프레임 (t-2)까지를 런으로 구성하고 증가되는 전송률로 보내야 하는 프레임 개수도 3에서 2로 감소된다.



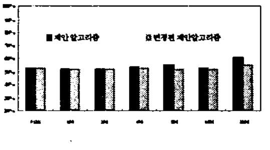
[그림 3] 제안 알고리즘 에 의한 [그림 2]의 전송 률 조절 예

4. 실험 결과

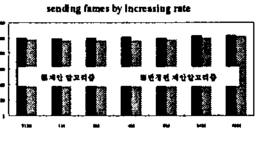
C언어로 스무딩 알고리즘을 구현하여 실험하였으며, 사용된 비디오 소스는 E.T 90이다[5,11]. 제안 알고리즘의 성능 평가를 위해 첨두 전송률, 전송률 변화량, 버퍼 이용률, 증가되는 전송률로 보내야 하는 프레임 개수를 비교하였다.

[그림 4]는 전송률 변화 횟수 비교로써, 이 값이 낮을 수록 서버와 클라이언트 간의 전송률 예약에 관한 통신 오버헤드가 적게 소요된다[7,9,10]. 제안 알고리즘의 전송률 변화 횟수가 높다. 이것은 제안된 알고리즘에서 런으로 검색되는 프레임들 중에서 전송률 증가가 가장 적은 전송률을 갖는 프레임까지를 런으로 설정하기 때문이다.

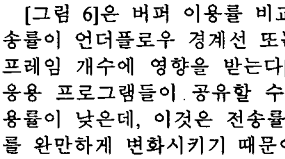
[그림 5]는 첨두 전송률 비교이다. 이 값이 낮을수록 서버와 클라이언트 간의 네트워크 자원들이 다수의 비디오 스트림을 서비스 할 수 있다[7,9,10]. 제안 알고리즘의 첨두 전송률이 버퍼 크기가 4M에서 낮고 이를 제외하면 같다. 이것은 제안 알고리즘에서 전송률의 증가가 필요한 경우에 전송률 증가량을 최소로 하는 프레임을 검색하기 때문이다.



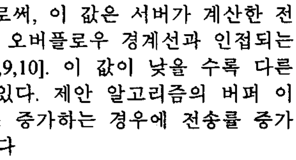
[그림 4] 전송률 변화 횟수 비교



[그림 5] 첨두 전송률 비교



[그림 6] 버퍼 이용률 비교



[그림 7] 증가되는 전송률로 보내는 프레임 개수 비교

[그림 6]은 버퍼 이용률 비교로써, 이 값은 서버가 계산한 전송률이 언더플로우 경계선 또는 오버플로우 경계선과 인접되는 프레임 개수에 영향을 받는다[7,9,10]. 이 값이 낮을 수록 다른 응용 프로그램들이 공유할 수 있다. 제안 알고리즘의 버퍼 이용률이 낮은데, 이것은 전송률이 증가하는 경우에 전송률 증가를 완만하게 변화시키기 때문이다

[그림 7]은 증가되는 전송률로 보내야 하는 프레임 개수 비교이다. 이 값이 낮을수록 짧은 시간 동안에만 증가되는 전송률이 사용되기 때문에 네트워크 자원의 효율성이 높아질 수 있다. 제안 알고리즘이 증가되는 전송률로 보내야 하는 프레임 개수가 적다.

5. 결론 및 추후 연구 방향

EBA 알고리즘에서는 e-PCRTT 알고리즘의 문제점을 개선하였지만 전송률 증가량을 고려하지 않기 때문에 급격하게 높은 전송률이 요구될 수 있으며 전송률이 증가 또는 감소되는 경우에 동일한 방법으로 전송률을 설정하기 때문에 증가되는 전송률로 보내야 하는 프레임 개수가 많아질 수 있다. 이는 네트워크 자원의 효율성을 감소시킬 수 있다.

따라서, 본 논문에서는 EBA 알고리즘의 문제점을 개선하여 전송률의 증가가 필요한 경우에는 검색되는 프레임들 중에서 가장 완만하게 전송률이 증가될 수 있는 프레임을 선택하여 이 프레임을 런의 마지막 프레임으로 설정하는 알고리즘을 제안하였다. 제안 방법의 성능은 전송률 변화 횟수, 첨두 전송률, 버퍼 이용률, 증가되는 전송률로 보내야 하는 프레임 개수의 평가 요소들을 사용하여 EBA 알고리즘과 비교하였다. 제안 알고리즘은 전송률 변화 횟수는 많지만 첨두 전송률, 버퍼 이용률, 그리고 증가되는 전송률로 보내야 하는 프레임 개수 비교에서는 우수하였다. 추후에는 이 제안 방법을 인터넷 상황에서 적용할수 있도록 연구할 예정이다.

참고 문헌

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

[2] W. Feng, F. Jahanian, S. Sechrest, "An Optimal Bandwidth Allocation Strategy for the Delivery of Compressed Prerecoded Video",ACM/Springer-Verlag Multimedia Systems, vol. 5, no. 5, pp. 297-309, Sept 1997.

[3] 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.

[4] 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.

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

[6] 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.

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

[8] 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.

[9] 이면재, 박도순, "인터넷상에서 효율적인 멀티미디어 서비스를 위한 전송률 조절 알고리즘", 한국정보처리학회 학술발표 논문집,2004.10.

[10] 이면재, 이준용, 박도순,"가변 비트율 비디오 전송을 위한 효율적인 스무딩 알고리즘",한국정보처리학회논문지 C, 2004.12.

[11] http://www.cis.ohio-state.edu/~wuchi.