# 서버 작업 부하를 고려한 개선된 주기적 멀티캐스트

한종현° 박승규
아주대학교 정보통신전문대학원
{hanbell°, sparky}@ajou.ac.kr

## A Periodic Multicast Strategy to Reduce Server Workload

Jonghyun Han° Seungkyu Park
Graduate School of Information and Communication, Ajou University

요 약

멀티미디어 서비스는 미디어 오브젝트의 대용량 특성으로 인해 클라이언트에게 서비스를 수행할 때 QoS를 보장하기 어렵다. 이러한 문제를 해결하기위한 많은 연구가 진행되었는데, 이중 사용자의 요구에 대한 적절한 응답시간과 서버의 부하를 줄이기 위한 목적으로 주기적인 멀티캐스트가 제안되었다. 하지만 오브젝트를 세그먼트로 나누는 이 멀티캐스트 정책은 세그먼트 개수의 증가에 따라 서버의 작업부하가 커지게 되는 단점을 가진다. 따라서 본 논문에서는 주기적인 멀티캐스트를 수행하는 서버의 작업부하를 줄이기 위해 스카이스크래퍼 멀티캐스트와 하모닉 멀티캐스트를 통해 서버의 휴지시간을 유도하고, 이에 따라 서버가 멀티캐스팅하는데 이용되는 시간을 감소시키는 멀티캐스트 정책을 제안한다.

### 1. 서론

멀티미디어 콘텐츠의 서비스에서는 멀티미디어 데이터의 대용량 특성으로 인해 발생되는 여러 가지 제한요소들이 고려되어야한다. 이런 제한 요소를 해결하기 위한 방법으로 클라이언트에게 지연 및 지터가 적은 서비스를 제공하고, 서버의 부하를 감소시키기 위해 프락시가 도입되었으며, 서버 및 네트워크 부하를 감소효과를 얻기 위해 멀티캐스트가 연구되어왔다. 멀티캐스트는 기본적으로 서버가 주기적으로 미디어 세그먼트를 방송하고, 클라이언트는 수신한 세그먼트를 조립하여 이를 재생하는 방법을 이용한다.

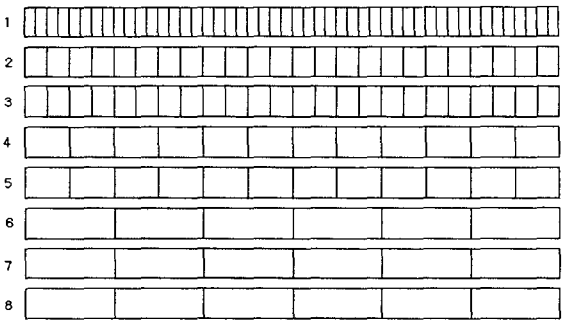
그러나 각 세그먼트를 주기적으로 방송을 해야 하는 서버는 인기 있는 비디오가 주가 되는 멀티캐스트를 수행하는 경우에도 많은 채널과 넓은 대역폭이 요구된다. 이에 따라 서버의 멀티캐스팅에 따른 부하를 줄이는 것은 시스템의 확장성 및 성능에 많은 영향을 미친다고 할 수 있다. 따라서 본 논문에서는 세그먼트 방송 스케줄 조정을 통한 주기적 멀티캐스팅에서의 서버 부하 감소 정책을 제안한다.

### 2. 관련연구

주기적인 방송에서는 멀티미디어 오브젝트를 길이 단위의 세그먼트로 나누어 각 세그먼트를 일정한 주기로 방송한다. 이때 서버는 클라이언트가 오브젝트 세그먼트를 다운받아 재생하는데 무리가 없는 수준의 세그먼트 방송을 수행해야 한다.

주기적 방송 기법에는 대표적으로 스카이스크래퍼 모양으로 오브젝트를 나누는 방식과 일정한 길이로 오브젝트를 나누고 하모닉하게 방송을 하는 방식이 있다.

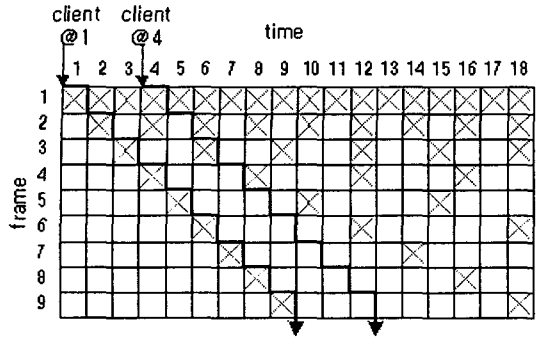
스카이스크래퍼 방송은 오브젝트의 앞부분은 짧은 길이로, 그리고 스트림의 후반부로 갈수록 점차 길게 오브젝트를 나누어 초기 지연시간을 줄이고, 또한 클라이언트가 사용하는 대역폭을 줄여주는 방식이다.[1]




[그림 1] 스카이스크래퍼 멀티캐스트

하모닉 멀티캐스트는 미디어를 일정한 크기의 세그먼트로 분할한다. 분할된 첫 번째 세그먼트는 지속적으로 방송하고, 두 번째 세그먼트는 한 개의 세그먼트 길이만큼 쉬는 시간을 갖고 방송하며, 세 번째 세그먼트는 두 개의 세그먼트 길이만큼 쉬는 시간을 갖고 멀티캐스팅을 하는 방법으로 초기지연시간이 있을 수 있지만, 서버의 부하는 스카이스크래퍼 멀티캐스트보다 낮다.[2]

그러나 이들 서버는 서버의 부하로 인해 동일한 수의 멀티미디어 오브젝트를 멀티캐스트하더라도 멀티캐스팅하는 방식에 따라 소모되는 대역폭의 차이가 클 뿐만 아니라 초기지연 해결방법을 제시해야하는 문제가 따른다. 따라서 본 논문에서는 이러한 주기적 멀티캐스팅 작업이 수행되는 서버의 과부하 문제의 해결을 위해 멀티캐스팅 스케줄 조절에 의한 서버 작업 부하 감소 정책을 제안하고, 제안된 방식을 통해 서비스 되는 경우 서버의 부하의 변동과 사용자 버퍼의 요구량을 살펴본다.




[그림 2] 하모닉 멀티캐스트

### 3. 시스템 구조

시스템은 크게 서버, 프락시 서버 및 클라이언트의 구성되어 멀티미디어 서비스를 수행한다.

#### 3.1 프락시 서버

서버는 서비스의 QoS보장을 위해 클라이언트에서 서버로부터 전달받은 스트림을 재생할 때, 될 수 있는 한 재생에 끊김이 없도록 그리고 사용자 요구 발생 후 재생까지의 초기지연시간을 짧게 유지할 수 있도록 하는 멀티캐스팅 정책을 이용해야한다.

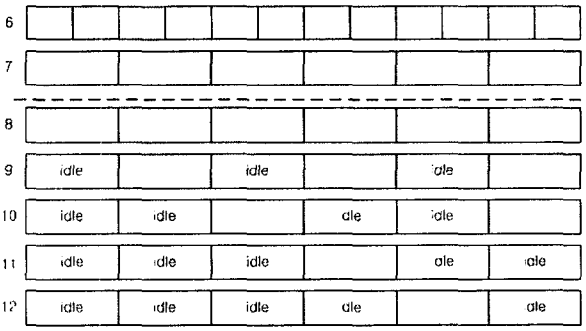
우선 사용자의 요구에 대한 초기 지연을 감소시키기 위해 사용자 인근에 프락시 서버를 두고 이를 사용자 서비스를 위한 prefix 캐쉬로 이용한다. 프락시 서버는 미디어 스트림의 각 세그먼트에 대한 prefix를 캐시하고 있으며, 클라이언트가 미디어 오브젝트를 요청하면, 프락시 서버는 갖고 있는 prefix를 통해 사용자에게 바로 서비스를 수행하고, 클라이언트는 프락시 서버로부터 전달받은 prefix를 재생하는 동안 서버로부터 suffix에 대한 버퍼링을 수행한다. 이러한 구성은 사용자의 요청에 대한 초기지연 감소 및 일정수준의 서버 부하 감소 효과를 기대할 수 있도록 한다.[4]

#### 3.2 서버

스카이스크래퍼 멀티캐스트의 방법은 사용자의 초기지연 감소를 주목적으로 하는 멀티캐스팅 정책이다. 즉, 오브젝트의 초반부를 구성하는 세그먼트의 크기를 작게 하여 이의 전송에 소요되는 지연을 줄이는 방식이다. 이후 오브젝트는 크기는 점차 큰 사이즈의 세그먼트로 분할되며, 최종 세그먼트의 크기는 클라이언트의 버퍼수준을 고려하여 적정수준의 크기로 결정된다.[1] 그러나 스카이스크래퍼 멀티캐스트는 미디어 오브젝트를 많은 수의 세그먼트로 나눌 수밖에 없다는 단점을 가진다. 이에 따라서 멀티캐스팅을 수행해야 하는 채널이 증가하게 되고, 이것은 바로 서버 작업부하의 증가를 초래한다. 그러나 오브젝트 후반부에 대해 적용되는 동일크기의 세그먼트 분할은 이 부분에 대해 하모닉 멀티캐스트가 응용될 수 있도록 한다. 이는 클라이언트의 요청이 미디어 오브젝트의 중간에서는 일어날 수 없다는 가정 아래서 가능하다.

제안하는 멀티캐스트 알고리즘은 스카이스크래퍼 멀티캐스트 기반에서 볼 때, 서버에게 유휴 시간을 부여하여 서버의 부하를 줄여줄 수 있지만, 동시에 더 많은 채널에서 세그먼트를 다운받도록 한다. 즉, 스카이스크래퍼 알고리즘에서는 최소 홀수 채널과 짝수 채널 하나씩 모두 두 개의 채널에서 세그먼트를 다운받는 것으로 오브젝트를 무리 없이 재생할 수 있다. 그러나 제안하는 알고리즘에서는 하모닉 멀티캐스트 알고리즘으로 전환되는 시점에서 두 개의 채널에서 다운되는 세그먼트로는 방송을 제대로 재생할 수 없게 된다. 각기 다른 주기로 휴식시간을 갖고 세그먼트가 방송되기 때문에 세그먼트를 요청하는 시기에 서버에서 멀티캐스팅을 수행해야 한다. 따라서 클라이언트는 동시에 방송되는 세그먼트를 모두 다운받아야 한다.

하모닉 멀티캐스트 알고리즘과 비교해볼 때, 제안하는 멀티캐스트 정책은 초기지연이 줄일 수 있고, 클라이언트의 채널사용 수는 줄어들지만, 서버에서 방송해야하는 세그먼트 수가 많아진다. 따라서 서버 부하의 경우 평균적으로는 하모닉 멀티캐스트 알고리즘이 제안하는 알고리즘보다 좋은 성능을 보이나, 하모닉 멀티캐스팅 정책의 경우 클라이언트 측에서는 상황에 따라 현재 서비스되고 있는 모든 채널에서 세그먼트를 받아 버퍼링을 수행해야하는 경우가 발생하므로, 버퍼의 사용효율이 저하된다.




[그림 3] 제안된 알고리즘으로 방송하는 서버

[그림 3]은 제안된 알고리즘에 따른 멀티캐스팅을 나타내고 있다. 제안된 알고리즘에서는 1~7번 채널까지 스카이스크래퍼 멀티캐스트를 따르고, 8번 채널부터 마지막 채널까지는 세그먼트 크기가 일정하므로 하모닉 멀티캐스트를 따른다. 이에 따라 9번 채널부터는 세그먼트의 주기에 따라 서버의 유휴시간이 발생한다. 따라서 서버에서는 동시에 서비스를 수행하는 채널이 줄게 되고, 이는 서버의 작업부하 감소를 유발한다.

#### 3.3 클라이언트

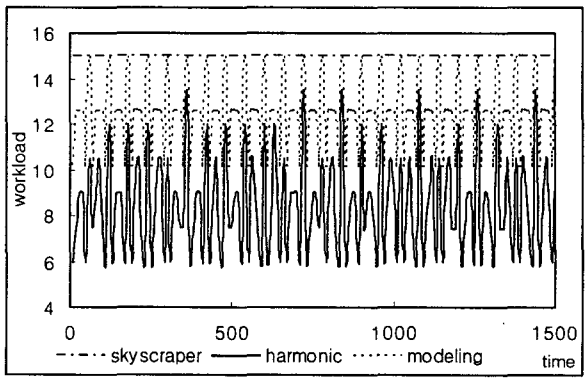
초기지연 감소를 위한 스카이스크래퍼 방식의 상위 채널에서 분할되는 세그먼트의 수가 증가하므로 서버에서

멀티캐스팅하는 세그먼트의 수도 따라서 증가하게 되나, 하모닉 멀티캐스팅에서와 같이 클라이언트가 멀티캐스팅 되는 모든 채널에서 세그먼트를 다운받을 필요는 없다. 이는 스카이스크래퍼 부분은 같은 내용의 세그먼트를 반복적으로 방송을 하고 있기 때문에, 필요한 시기에만 다운받으면 되기 때문이다. 그러나 하모닉 멀티캐스트를 따르는 채널에서는 재생하고 있는 채널부터 현재 서버가 멀티캐스트하는 모든 채널에서 다운을 받는다. 따라서 제안된 알고리즘을 사용하기위해서는 클라이언트에서 적정 크기의 버퍼가 요구된다.

스카이스크래퍼 멀티캐스트 구간에서는 한 세그먼트의 크기가 하모닉 멀티캐스트 구간에서의 한 세그먼트 크기보다 작으므로 클라이언트 버퍼의 크기는 하모닉 멀티캐스트 구간의 채널 수에 따라 달라진다.

## 4. 시뮬레이션

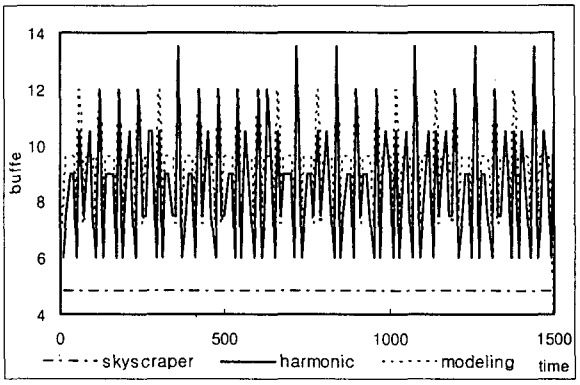
비디오 길이는 15라고 가정하였고, 스카이스크래퍼 멀티캐스트에서 세그먼트는 [0.2, 0.4, 0.4, 1, 1, 2.4, 2.4, 2.4, 2.4, 2.4]로 하였다. 그리고 하모닉 멀티캐스트에서 세그먼트는 길이를 1.5라고 하고 두 가지의 경우 모두 총 10개의 채널을 사용한다고 하였다.




[그림 4] 서버의 작업 부하

[그림 4]는 각 정책별 서버의 작업 부하 비교를 보이고 있다. 이 실험에서는 스카이스크래퍼 멀티캐스트의 서버 작업 부하가 가장 크게 나타났으며, 평균적으로 제안된 모델에서 스카이스크래퍼 정책의 50% 정도의 작업부하가 나타났다. 이것은 서비스 후반부에 제안된 정책에 따라 서버 유휴기간이 길어져 서버의 작업 부하가 줄어든 결과이다. 여기에서 제안된 정책이 하모닉 멀티캐스트 정책보다 부하가 크게 나온 것은 사용하는 채널의 총 개수는 동일하더라도 서비스를 수행하는 동안에 요구되는 채널의 수가 하모닉 멀티캐스팅 정책보다 많기 때문이다. 그러나 하모닉 멀티캐스팅은 클라이언트의 버퍼크기와 사용률을 고려하지 못한다.

[그림 5]는 서비스에 사용되는 클라이언트의 버퍼 사용량을 정책에 따라 비교하고 있다. 제안된 정책은 기존의 정책에 비해 클라이언트의 버퍼 사용량이 늘어났음을 보인다. 이것은 스카이스크래퍼 멀티캐스팅 서버의 작업부하를 줄이기 위해 휴지시간을 줌으로써 캐쉬 해야 하는 세그먼트가 늘어났기 때문이다. 그리고 평균값은 하모닉 멀티캐스팅보다 높게 나왔지만, 표준편차는 오히려 더 낮아서 버퍼의 사용 효율이 더 좋아졌다.




[그림 5] 클라이언트의 버퍼 사용량

## 5. 결론 및 향후계획

본 논문에서는 서버의 작업 부하를 줄이기 위한 정책으로 주기적 멀티캐스트에서 세그먼트 길이에 따른 멀티캐스트 스케쥴링 정책을 제시하였으며, 그 결과로 서버의 유휴시간 발생에 따라 서버의 부하가 감소하는 것으로 나타났다. 이러한 서버의 유휴시간은 다른 비디오에 대한 멀티캐스트에 사용할 수 있게 되고, 서버의 확장성을 증가시키는 효과를 얻을 수 있다. 그러나 서버의 작업 부하가 줄어드는 만큼 클라이언트에서 버퍼의 사용 상대적으로 부하가 늘어나게 되었다. 이에 따라 클라이언트에서의 효율적인 버퍼사용연구가 필요하다.

## 6. 참고문헌

[1] Kien A.Hua and Simon Sheu, "Skyscraper broadcasting: A new broadcasting scheme for metropolitan video-on-demand service ", in Proceedings of IEEE Infocom 2001

[2] Li-Shen Juhn and Li-Meng Tseng, "Harmonic broadcasting for video-on-demand service", IEEE Transactions on Broadcasting, vol. 43, no. 3 pp 268-271, Sept, 1997

[3] A Multi-Multicast Sharing Technique for Large-Scale Video Information Systems, ICC 2002, IEEE, April-May 2002

[4] S.Sen, J.Rexford, and D.Towsley, "Proxy prefix caching for multimedia streams", in Proc. IEEE INFOCOM, April 1999