

WiFi 환경에서 적응적 HTTP 스트리밍을 위한 프록시 기반의 공정하고 안정적인 전송을 적응 방안

권윤, 임영빈, 최준혁 권태경*
서울대학교 컴퓨터 공학부

{ykwon, ybim, jhchoi}@mmlab.snu.ac.kr, *tkkwon@snu.ac.kr

Proxy based fair and stable rate adaptation for HTTP adaptive streaming WiFi networks

Yoon Kwon, Youngbin Im, Jun-hyuk Choi, Ted "Taekyoung" Kwon
Schol of Computer Science and Engineering, Seoul National University

요 약

최근 HTTP 기반의 적응형 스트리밍 기술은 폭발적으로 성장하는 비디오트래픽의 양과, 그에 따른 사용자 QoE(Quality of Experience)에 대한 요구의 증가로 인해 큰 주목을 받고 있다. 이러한 HTTP 기반의 적응형 스트리밍 기술을 DASH(Dynamic adaptive streaming over HTTP)라 하는데 DASH 클라이언트는 매 패킷단위로 가용 네트워크 대역폭을 측정하고, 이러한 정보를 기반으로 다운로드 받을 비디오 세그먼트의 비트레이트를 결정할 수 있다. 그러나 각 클라이언트에 대한 리소스 블록 정보를 알 수 있는 LTE의 환경과는 달리, Wifi 환경에서는 이러한 정보를 알 수가 없다. 따라서 본 논문에서는 AP와 서버 사이에 프록시를 두어, 오고 가는 패킷을 캡처하여 네트워크 상황에 대한 정보를 클라이언트에 전달함으로써 Wifi 환경에서 보다 사용자에게 맞도록, 보다 공정(fair)하고 안정(stable)적인 전송을 적응(rate adaptation)을 할 수 있는 방안에 대해 보여주하고자 한다.

1. 서 론

모바일 기기의 폭발적인 증가와 이동통신 기술의 발달로 멀티미디어 스트리밍 서비스는 전 세계적으로 가장 인기 있는 서비스 중 하나가 되었다. Cisco에서 발표한 자료에 따르면, 비디오 트래픽은 전체 인터넷 트래픽에서 가장 높은 비율을 차지하고 있으며, 트래픽 양은 점점 더 증가하고 있다.

이처럼 인터넷이 가능한 디바이스가 증가함에 따라 디바이스 플랫폼에 관계없는 인터넷 프로토콜인 HTTP(Hyper text transport protocol) 기반의 비디오 스트리밍에 대한 연구가 진행되었고, 청크 기반의 HTTP 통신에서 사용자들에게 높은 체감 품질을 어떻게 보장을 할 것인지가 핵심 이슈로 부상하였다. 이를 위해 제안된 여러 기술들 중 DASH는 이러한 사용자 QoE를 만족 시킬 수 있는 가장 대표적인 기술이다. DASH는 MPEG(Moving picture experts group)에서 표준화한 적응적 HTTP 스트리밍 기술로, DASH 서버에서는 비디오 콘텐츠를 각기 다른 비트 레이트를 갖는 여러 버전으로 인코딩 한 다음, 인코딩된 비디오 데이터를 2초 정도의 작은 세그먼트 단위로 분할해 둔다. 이 세그먼트들이 DASH 클라이언트의 요청에 따라 전달이 되는데, DASH 클라이언트에서는 자신의 상황에 따라서, 비트 레이트 적용 알고리즘을 수행하여 네트워크의 가용 대역폭을 계산하고 그 결과를 토대로 다음 비디오 세그먼트의 비트 레이트를 결정한다. 즉, DASH 클라이언트의 비디오 플레이어가 현재

단말의 네트워크 상태 및 QoE를 고려하여, 적절한 비디오의 화질을 선택하여 서비스 하도록 한다는 것이다. 따라서 Wifi 환경에서는 AP에 연결된 단말들이 최대한 공정하고 안정적으로 서비스를 받는 것이 중요한데, Wifi 환경에서는 LTE처럼 무선자원을 특정 단말에 정확하게 할당해 주는 것이 매우 힘들기 때문에 클라이언트 측에서 대역폭을 예측하는 DASH 알고리즘이 보다 견고해질 필요가 있다. 이에 따라, 우선 비디오 스트리밍 상황에서 단말별 전송 정보에 대한 데이터를 기록하고 해당 네트워크 상황 정보를 보내줄 프록시 서버를 AP 뒤에 두고, 최근 안정성과 효율성, 그리고 공정성 측면에서 좋은 결과를 보여준, Improving Fairness, Efficiency, and Stability in HTTP-based Adaptive Video Streaming with FESTIVE에서 제시한 알고리즘을 [1] 바탕으로 이를 실험 상황에 맞게 수정하여 둘 사이의 결과를 비교해 안정성과 공정성에 어떠한 향상이 있을지 살펴보았다.

2. 실험 환경

2.1 실험 시나리오

그림 1은 Wifi를 위한 DASH 알고리즘을 적용하기 위한 논리적인 실험 아키텍처로서, Wifi의 AP에 연결된 클라이언트들이 미디어 서버에 비디오 세그먼트를 요청하면, 미디어 서버와 AP 사이에서 전송되는 패킷을 모니터링

하는 프록시서버는 단말의 IP를 패킷으로부터 확인해서, 각 단말별 요청과 전송 상황을 기록하게 된다. 세그먼트 당 요청된 비디오 레이트의 EWMA를 계산하여 자신의 현재 레이트 값과 다른 단말들의 평균 사이의 비율을 응답 패킷에 기록하여 전달함으로써, 현재 네트워크 상황에서 자신의 비디오레이트의 상대적인 크기를 단말에서 알 수 있도록 하였다.

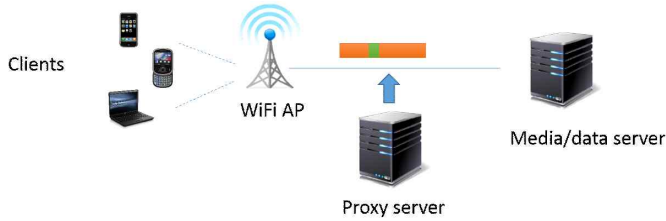


그림 1 Wifi를 위한 DASH 아키텍처

이 정보를 토대로 단말에서는 DASH 알고리즘을 통해 자신의 비디오 레이트를 올릴지 유지할지를 선택하여 전송율을 적응할 수 있다.

2.2 실험 설정

실험에서는 AP 한 대에 3대의 단말을 연결하였고, 미디어 서버와 프록시를 실험의 편의상 같은 머신에서 동작하도록 하였다. 이때, 프록시 서버의 동작을 위해서 libnetfilter queue를 사용하였으며 [2] 이를 통해 커널 레벨에서 패킷을 살펴보고 처리하도록 하였다. 미디어 데이터로는 오픈 라이선스로 제공되는 Sintel이라는 영상을 [3] 사용하였다. DASH 비디오 플레이어는 javascript 기반의 구글에서 공개한 MPEG-Dash player [4]의 소스코드를 수정하여, FESTIVE 알고리즘을 바탕으로 구현하였다. 실험에 사용된 비디오 레이트는 788KBps, 1323KBps, 2285KBps, 3188KBps, 4106KBps, 5020KBps, 5907KBps의 7개 레이어로 설정하였고 실험 결과 확인을 위해 AP의 최대 전송 대역폭을 10MBps로 제한하였다.

2.3 기존 알고리즘의 활용

기존의 FESTIVE 알고리즘에서는 효율성과 공정성 및 안정성의 균형을 맞추기 위해 비디오 레이트를 높일 때 레이어 1로부터 한 단계씩 높여가면서 당장 높일수 있다 하더라도 현재 비디오 세그먼트의 레이어 번호 만큼의 세그먼트를 요청하는 동안 기다렸다가 한 단계 올리는 방식의 보수적인 접근을 택하고 있다. 즉, Layer가 4라면 예측된 대역폭에 의해 더 높은 레이어가 선택되더라도, Layer 4의 요청을 4번 보내는 동안은 레이어를 올리지 않는다는 것이다.

$$R_{l+1} = R_l + l (l \geq 1, layer\ vumber)$$

수식 1. FESTIVE 알고리즘의 비디오 요청횟수

이러한 비디오 레이트 증가 함수를 본 논문에서는 상대적으로 높은 레이트의 단말에 대해서는 더 보수적으로 증가시키도록 하여 상대적으로 낮은 단말의 레이트가 증가 될 수 있도록 기다리도록 하였다.

$$\begin{cases} R_{l+1} = R_l + l (l \geq 1) & , \text{when } r \leq 1 \\ R_{l+1} = R_l + l * r * 2 (l \geq 1) & , \text{when } r > 1 \end{cases}$$

l : layer\ vumber, r : ratio of videorate

수식 2. FESTIVE를 활용한 해당 비디오 요청횟수

수식2에서 r 의 값은 프록시 서버에서 매 세그먼트 요청 시 마다 수집한 비디오레이트를 기준으로 자신을 제외한 다른 두 단말의 EWMA의 평균 값 대비 자신의 EWMA값의 비율이며, 수식3을 통해 계산하도록 하였다.

$$r_{n,1} = \frac{ewma_{n,1}}{(ewma_{n,2} + ewma_{n,3})/2}, n : n\text{-th segment request}$$

$$ewma_{n,k} = 0.9 \text{ videorate}_n + 0.1 ewma_{n,k-1}$$

수식 3. 비디오 레이트 비율 계산

따라서 수식3에서는 n 번째 세그먼트의 리퀘스트 상황에서의 비율은 그시점에서 2와 3 단말의 ewam 값의 평균분에 1단말의 ewma이고 ewma는 새로 들어온 videorate에 0.9의 가중치를 두도록 exponential weight moving average의 결과이다.

3. 실험 결과

그림 2와 그림 3은 각각 Festive 알고리즘만 적용 했을 때의 비트레이트 변화와 Wifi 적응 알고리즘을 적용했을 때의 변화이다. 그래프에서 가장 먼저 눈에 띄는 점은 레이트 변화 횟수이다. 2.3에서 설명한 바와 같이, 상대적으로 높은 레이트를 갖는 단말에 대해 더 보수적인 변화를 적용한 결과 레이트의 변화 횟수가 현저하게 줄어든 것을 알수 있다. 평균적으로 Festive의 결과에서는 평균 28회 레이트 변화가 일어난 반면, Wifi 적응 알고리즘을 사용한 경우, 평균 19.3회의 변화가 발생하여 약 30% 감소함을 확인할 수 있었다. 비디오 스트리밍에서 레이트 변화 횟수는 유저의 QoE에 상당한 영향을 끼친다는 것을 고려할 때 이러한 결과는 상당히 의미가 있다. 안정성의 측면에서도 동일한 레이어를 선택한 시간이 현저하게 높고, 한 단말이 다른 단말들의 레이트를 독점하는 현상이 적은 것을 그래프에서 살펴 볼 수 있다. 또한 비디오 재생기간 동안의 평균 비디오 레이트를 비교한 결과가 그림 4와 그림 5인데, 각 노드별 차이가 FESTIVE의 그것보다 훨씬 작게 나타나, 공평성의 측면에서도 기존의 알고리즘 보다 나은 결과를 나타냄을 확인할 수 있었다.

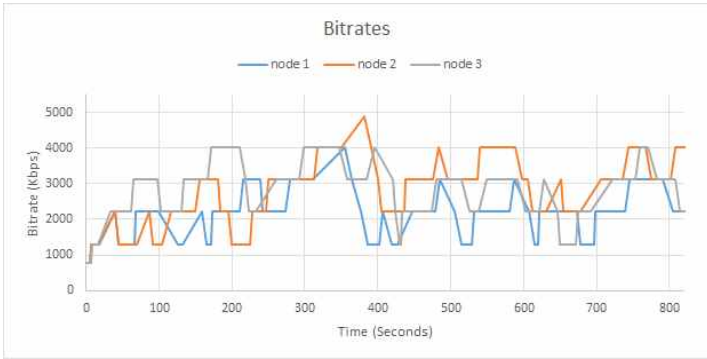


그림 2 Festive 알고리즘 비트 레이트 변화

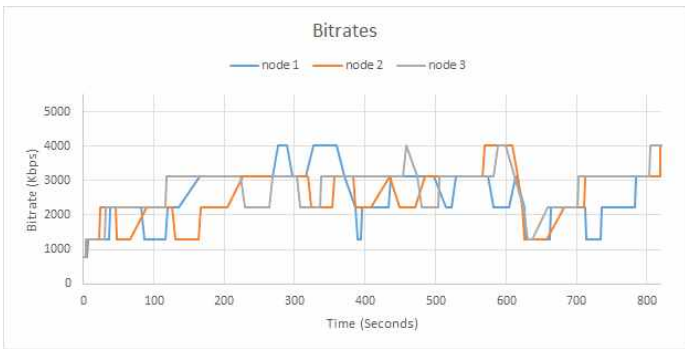


그림 3 Wifi Adaptation 알고리즘 적용시 레이트 변화

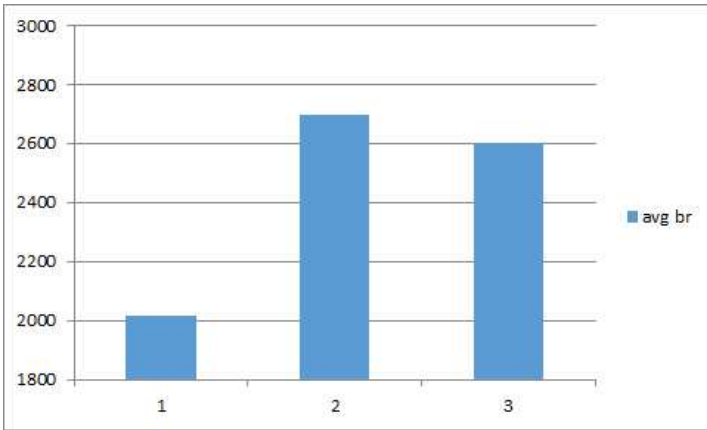


그림 4 FESTIVE 알고리즘에서 평균 비디오 레이트

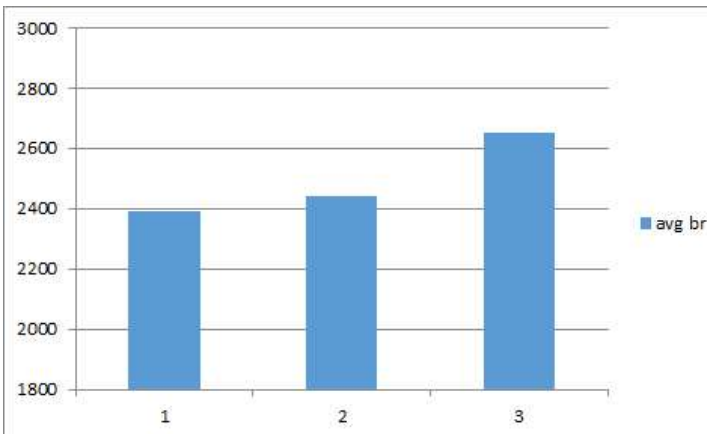


그림 5 Wifi 적응 알고리즘에서 평균 비디오 레이트

3. 결론 및 향후 연구

본 논문의 실험을 통해 우리는 Wifi 환경에서 프록시 서버를 통해 전달된 네트워크 정보를 토대로 적응 알고리즘을 수행하여 보다 공평하고 안정적인 스트리밍이 가능함을 확인할 수 있었다. 적응 알고리즘은 실제로 더 많은 실험이나 수학적 분석을 토대로 향상된 알고리즘을 발견할 수 있겠지만, 본 논문에서는 높은 레이트의 단말에 대해 상수배의 적응 알고리즘 수정을 통해서도 아이디어에 대한 검증이 가능하였다. 향후 연구해 볼 만한 주제로는 무선 상황에서 이동성에 대한 고려가 중요한데, 본 논문에서는 무선 상황이 공평함을 가정하였기 때문에, 링크상황의 변화와 이동성에 대한 연구가 더 필요할 것으로 생각된다.

ACKNOWLEDGMENT

본 연구는 미래창조과학부 및 정보통신기술진흥센터의 정보통신·방송 연구개발사업의 일환으로 수행하였음. [2014-044-011-003, 개방 제어 기반 분산구조 모바일코어 네트워크 기술 개발]

참고 문헌

- [1] Junchen Jiang, Vyas Sekar, Hui Zhang. Improving Fairness, Efficiency, and Stability in HTTP-based Adaptive Video Streaming with FESTIVE. In CoNEXT, 2012.
- [2] libnetfilter_queue project http://www.netfilter.org/projects/libnetfilter_queue
- [3] sintel.org <https://durian.blender.org/download/>
- [4] Google DASH player <http://dash-mse-test.appspot.com/dash-player.html>