

# DASH 기반 멀티미디어 스트리밍 서비스에서 사용자 체감품질 향상을 위한 비트율 적응 기법

서 동은\*, 장 인 선\*, 백 상 현<sup>o</sup>

## A Video Bitrate Adaptation Algorithm for DASH-Based Multimedia Streaming Services to Enhance User QoE

Dongun Suh\*, Insun Jang\*, Sangheon Pack<sup>o</sup>

### 요 약

Dynamic adaptive streaming over HTTP (DASH)<sup>[2]</sup>는 최근 주목받고 있는 스트리밍 기술이다. DASH 클라이언트는 연속적으로 가용 네트워크 대역폭을 측정하고, 이를 기반으로 향후 다운받을 비디오 세그먼트의 비트율을 결정한다. 본 고에서는 최소 재생 버퍼 레벨을 보존하여 가변적인 네트워크 환경에서도 사용자 체감 품질 저하를 야기하는 재생 끊김 현상, 재생 중 비디오 품질의 급격한 변화 등을 효율적으로 방지하는 비트율 적응 알고리즘인 QoE-enhanced adaptation algorithm over DASH (QAAD)를 제안한다. 또한 DASH 테스트베드 상에 QAAD와 재생 버퍼를 고려한 기존의 알고리즘 (즉, QDASH[9])을 구현하여 실험적으로 성능을 비교 및 분석하였다. 네트워크 환경이 급변하는 환경에서 실험을 수행한 결과, QDASH는 재생 끊김 현상 및 비디오 화질의 잦은 변화가 발생하는 반면, QAAD는 재생 끊김 없이 안정적인 비디오 화질을 제공하는 것을 확인할 수 있었다.

**Key Words** : HTTP adaptive streaming, bitrate adaptation algorithm, bandwidth estimation

### ABSTRACT

Dynamic adaptive streaming over HTTP (DASH)<sup>[2]</sup> is the most recent and promising technology to support high quality streaming services. In dynamic adaptive streaming over HTTP (DASH), a client consecutively estimates the available network bandwidth and decides the transmission rate for the forthcoming video chunks to be downloaded. In this paper, we propose a novel rate adaptation algorithm called quality of experience QoE-enhanced adaptation algorithm over DASH (QAAD), which preserves the minimum buffer length to avoid interruption and minimizes the video quality changes during the playback. We implemented a DASH test bed and conducted extensive experiments. Experimental results demonstrate that under fluctuating network conditions, QAAD provides seamless streaming with stabilized video quality while the previous buffer-aware algorithm (i.e., QDASH[9]) frequently changes the video quality and undergoes the interruption.

\* 본 연구결과의 일부는 International Conference on Information Networking (ICOIN) 2014에 발표되었습니다.<sup>[1]</sup>

\* 본 연구는 미래창조과학부 및 정보통신산업진흥원의 대학 ICT연구센터 지원사업 (NIPA-2014-H0301-14-1015), 서울특별시의 서울벨 연구소 사업 (WR080951) 및 LG연암재단 해외연구교수사업의 지원을 받아 수행되었습니다.

♦ First Author : Korea University School of Electrical Engineering, fever1989@korea.ac.kr, 학생회원

o Corresponding Author : Korea University School of Electrical Engineering, shpack@korea.ac.kr, 종신회원

\* 고려대학교 전기전자공학과, zerantoss@korea.ac.kr

논문번호 : KICS2014-05-184, Received May 10, 2014; Revised June 13, 2014; Accepted June 13, 2014

## I. 서 론

모바일 기기의 보편화, 디스플레이 기술의 발달 및 이동통신 기술의 발달로 인해 멀티미디어 스트리밍 어플리케이션은 인터넷 접속이 가능한 전자기기 (스마트폰, 태블릿 PC, IPTV 등)의 사용자들에게 가장 인기 있는 어플리케이션으로 자리매김 하였다. 한편, 멀티미디어 스트리밍 서비스를 제공하는데 있어, 사용자들에게 높은 사용자 체감 품질을 보장하는 것은 가장 중요한 과제임과 동시에 아직 완벽히 해결되지 않은 과제이다. 이를 위해, 종래에 여러 기술들 (네트워크 자원 예약, 연결 수락 제어, 적응적 스트리밍 기술 등)이 제안되어왔다. 이들 중, 적응적 스트리밍 기술은 사용자의 체감 품질을 만족시키는 가장 필수적인 기술이다. 적응적 스트리밍 기술은 네트워크 환경과 클라이언트 기기의 환경을 고려하여 자동으로 알맞은 비트율을 선택해 준다. 예를 들어, 적응적 스트리밍에서는 가용 대역폭과 클라이언트 기기의 디스플레이 종류에 맞게 비디오 콘텐츠의 해상도와 초당 프레임 수를 적응시킬 수 있다. 따라서 종래의 단일 비디오 비트율 스트리밍 기술 대비, 사용자 체감 품질에 직접적인 영향을 미치는 재생 지연 시간 및 재생 끊김 횟수를 단축시키는 장점이 있다. 한편, 네트워크 프로토콜 관점에서, 종래의 스트리밍 기술들은 user datagram protocol (UDP) 상에서 동작하는 real-time transport protocol (RTP)/real-time streaming protocol (RTSP)를 사용하였다. 하지만, 최근 상업형 스트리밍 솔루션들인 Microsoft의 Smooth Streaming, Apple HTTP Live Streaming 그리고 Adobe HTTP Dynamic Streaming 외에 대다수의 스트리밍 서비스에서는 hyper text transport protocol (HTTP)가 사용된다. 그 이유는 HTTP가 기존의 RTP에 비해 다음의 장점들을 갖기 때문이다. 첫째, 기존의 HTTP 서버 및 캐쉬를 재사용 할 수 있다. 둘째, HTTP는 transport control protocol (TCP) 상에서 동작하기 때문에, UDP 상에서 동작하는 RTP를 사용할 경우 생기는 문제인 network address translation (NAT) 통과 문제가 발생하지 않는다.

Dynamic adaptive streaming over HTTP (DASH)<sup>[2]</sup>는 moving picture experts group (MPEG)에서 표준화한 가장 대표적인 HTTP 기반의 적응적 스트리밍 기술이다. DASH 서버는 하나의 비디오 콘텐츠를 각기 다른 비트율 (즉, 화질)을 가지는 복수의 버전으로 인코딩하고, 각 버전을 2-4 초의 세그먼트로 분할한다. 각 세그먼트는 HTTP 객체의 형태로

DASH 클라이언트에게 제공된다. DASH 클라이언트의 비트율 적응 알고리즘은 연속적으로 가용 네트워크 대역폭을 측정하고, 이를 기반으로 향후 다운받을 비디오 세그먼트의 비트율을 결정한다. 이 때, DASH 클라이언트의 비트율 적응 알고리즘은 사용자 체감 품질을 감소시키는 요인들 (즉, 재생 끊김, 급격한 비디오 화질의 감소, 그리고 재생 중 빈번한 비디오 화질 변화)을 최소화 하도록 비트율을 선택하여야 한다. 예를 들어, 가용 네트워크 대역폭이 감소하였을 때, 일전에 요청하던 비트율의 세그먼트를 다시 요청할 경우, 재생 버퍼 고갈로 인한 재생 끊김이 발생할 수 있으므로, 그에 맞춰 더 낮은 비트율을 가지는 세그먼트를 요청하는 것이다. 또한 재생 화질이 급격히 변하는 경우, 사용자 체감품질에 부정적인 영향을 미치기 때문에, 비디오 비트율의 변화를 최소화 하여야 한다<sup>[9]</sup>. 가용 대역폭이 급격히 변하는 네트워크의 경우, 이러한 요구사항을 만족시키는 것은 상당히 어려운 문제가 된다. DASH 표준 관점에서 이러한 비트율 적응 알고리즘은 구현 이슈에 속하기 때문에, 이에 대한 몇몇의 연구가 수행되었다<sup>[3-9]</sup>.

먼저 비트율 적응 알고리즘에 대한 성능 평가 및 분석을 수행한 연구들이 있다<sup>[3,4]</sup>. 실험을 통한 성능 평가를 수행한 연구에서는 상업형 적응적 스트리밍 솔루션들인 Netflix client, Microsoft Smooth Streaming, 그리고 Adobe OSMF에서의 비트율 적응 알고리즘에 대해 성능 평가를 수행하였다<sup>[3]</sup>. 실험 결과는 이들 중 어떤 알고리즘도 안정적인 비트율 결정 및 재생 버퍼 레벨의 안정화를 달성하지 못함을 보여 준다. 수학적 모델링을 통해 성능 분석을 수행한 논문에서는 비트율 적응 알고리즘의 동작을 상미분 방정식을 통해 모델링하여, 사용자의 체감 품질에 대한 분석적 모델을 제안하였다<sup>[4]</sup>.

몇몇 연구에서는 비트율 적응 알고리즘을 직접 개발하였다<sup>[5-9]</sup>. 수학적 도구를 활용하여 비트율 선택 기법을 제안한 연구에서는 오프라인 환경 (즉, 시간에 따라 변하는 가용 네트워크 대역폭이 주어진 환경)에서 Markov Decision Process (MDP)를 이용하여 최적의 비트율 선택 기법을 도출하였다<sup>[5]</sup>. 하지만, 실제 온라인 환경에서는 가용 네트워크 대역폭에 대한 정확한 예측이 힘들기 때문에, 이러한 기법은 성능 저하가 심각할 것으로 예상된다. 가변적인 네트워크 상황을 실시간으로 고려한 비트율 적응 알고리즘을 제안한 연구들에서는 가용 네트워크 대역폭보다 낮은 비트율을 결정함으로써 재생 버퍼의 고갈을 방지하고자 하였다<sup>[6,7]</sup>. 하지만 가용 네트워크 대역폭의 변동이

심할 경우, 이러한 알고리즘은 빈번하게 비트율을 바꾸기 때문에 사용자 체감 품질에 저하를 주는 문제점이 있다. 한편, 재생 버퍼를 보존하는 비트율 적응 알고리즘을 제안한 연구에서는 가용 대역폭이 충분히 높은 상황에서는 비디오 비트율을 보존적으로 한 단계씩 상승시키고, 가용 네트워크 대역폭이 낮은 경우, 비디오 비트율을 가용 대역폭 이하로 낮춤으로써 재생 버퍼 레벨을 항상 일정 레벨 이상으로 유지하였다<sup>18</sup>. 그러나 가용 대역폭이 급격히 감소할 경우, 해당 알고리즘은 급격히 비트율을 낮추기 때문에, 사용자 체감 품질을 저하시킨다. 이러한 문제를 완화시키는 비트율 적응 기법이 제안되었다<sup>19</sup>. 구체적으로, 가용 네트워크 대역폭이 급격하게 감소할 시, 임의의 비트율 선택에 대한 향후의 비디오 재생 버퍼 레벨을 예측하여, 재생 버퍼가 고갈되지 않는 조건 하에서 바로 이전에 선택하였던 비트율보다 한 단계 낮은 비트율을 선택한다. 이러한 비트율 선택은 남은 재생 버퍼량이 충분 할 경우에만 가능하다. 하지만, 해당 알고리즘은 가용 네트워크 대역폭이 급격하게 증가/감소할 경우 재생 버퍼 레벨의 관리를 효율적으로 하지 못하여, 급격한 비디오 화질 저하를 초래한다. 결론적으로, 앞서 언급한 어떤 비트율 적응 알고리즘도 네트워크 상황이 급변할 경우, 재생 끊김을 예방할 수 없었다.

한편, 무선 환경에서는 대역폭이 더욱 가변적이고, 채널 페이딩으로 인한 비디오 패킷 손실이 빈번하다. 이로 인한 재생 끊김 현상을 최소화하고자, 네트워크 코딩 기법을 가변적인 네트워크 상황 (비디오 패킷 손실률 등)에 따라 가변적으로 적용한 기법들이 제안되었다<sup>10-12</sup>. 본 연구에서는 패킷 손실에 의한 사용자 체감 품질 감소 효과를 고려하고 있지 않으나, 이는 무선 환경에서 DASH의 성능을 향상시킬 요소 기술이 될 수 있으므로 향후 연구 주제로 적합하다고 판단된다.

본 고에서는 사용자 체감 품질을 향상시키는 DASH에서의 비트율 적응 알고리즘 QoE-enhanced Adaptation Algorithm over DASH (QAAD)를 제안한다. 구체적으로, 제안하는 비트율 적응 알고리즘은 재생 버퍼를 효율적으로 관리하여, 가용 네트워크가 급변하는 상황에서도 재생 끊김, 급격한 비디오 화질 저하, 불필요한 비디오 화질 변화를 방지함으로써 사용자 체감 품질을 향상시킨다. 또한 DASH 클라이언트 (DASH VLC Plugin<sup>13</sup>)와 서버 (Apache HTTP 서버<sup>14</sup>)로 구성되는 DASH 테스트베드를 구축하여, 제안하는 알고리즘을 기존의 재생 버퍼를 고려한 비트율 적응 알고리즘인 QDASH<sup>19</sup>와 비교하여 성능 평가를 수행하였다. 실험을 수행한 결과, 가용 네트워크

대역폭이 주기적으로 크게 변하는 네트워크 상황에서 QDASH에서는 요청하는 비디오 비트율이 크게 변동하며 재생 버퍼 고갈로 인한 재생 끊김이 발생하는 반면, QAAD에서는 요청 비디오 비트율이 안정적이고 재생 버퍼 레벨이 안정적으로 유지되어 재생 끊김이 방지되는 것을 확인할 수 있었다.

서론에 이어, II장에선 제안하는 비트율 적응 알고리즘인 QAAD에 대해서 설명하고, III장에서는 실험을 통한 성능 평가를 수행한다. IV장에서는 본 고의 결론을 도출한다.

## II. 제안하는 DASH에서의 비트율 적응 알고리즘 (QAAD)

### 2.1 시스템 모델

그림 1은 본 고에서 제안하는 기법의 시스템 모델을 나타낸다. 또한 표 1은 중요한 변수를 요약한 표이다.  $L = \{1, 2, \dots, n\}$ 은 DASH 서버가 제공하는 비디오 품질 레벨의 집합을 나타낸다. 비디오 품질 레벨  $l$ 이 요구하는 비트율은  $b(l)$ 로 나타낸다. 예를 들어, 비디오 품질 레벨  $l+1$ 은 비디오 품질 레벨  $l$ 보다 높은 비트율을 요구한다 (즉,  $b(l+1) > b(l)$ ). 비디오 품질  $l$ 은  $m$ 개의 세그먼트로 구성되며 이는  $s_{l,1}, s_{l,2}, \dots, s_{l,m}$ 로 표현된다. 또한 모든 세그먼트는  $\tau$  초의 재생 시간을 갖는다고 가정하였다.

DASH 클라이언트의 비트율 적응 알고리즘은 1) 가용 대역폭 측정 기법 (Bandwidth Estimation), 2) 비트율 선택 기법 (Bitrate Selection)으로 구성된다. 가용 대역폭 측정 기법에서는 가용 대역폭  $bw_{estimated}$ 을 측정하고 이를 비트율 선택 기법에게 알린다. 비트율 선택은 세그먼트의 다운로드가 끝난 시점에서 수행된다. 해당 시점을  $t$ 라 하면, 비트율 선택 기법은 가용 네트워크 대역폭  $bw_{estimated}$  및 현재 재생 버퍼 길이  $B(t)$ 를 기반으로 다음 세그먼트의 비디오 품질

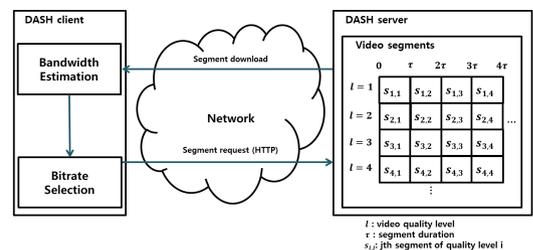


그림 1. 시스템 모델  
Fig. 1. System model

표 1. 변수 요약  
Table 1. Summary of notations

Notation	Definition
$\mu$	Pre-defined marginal buffer length.
$\sigma$	Pre-defined minimal buffer length.
$\tau$	Segment duration in second.
$bw_{estimated}$	Estimated bandwidth.
$B(t)$	Buffer length in video second at time $t$ .
$L$	Set of the video quality levels.
$l_{best}$	Video quality level not exceeding the available bandwidth.
$l_{prev}$	Video quality level of the previously requested segment.
$l_{next}$	Video quality level of the segment to be downloaded.
$b(l)$	Bitrate of the quality level $l$ .
$n_l$	The number of the segments which can be downloaded without interruption for the quality level $l$ .

$l_{next}$ 를 결정한다. 한편, 비디오 비트율이 가용 네트워크 대역폭을 넘지 않는 비디오 품질 중, 가장 높은 비디오 품질을  $l_{best}$ 라 표현하고, 이는 다음 수식으로 정의한다.

$$l_{best} = \arg \max_{l \in L} (b(l) \leq bw_{estimated}). \quad (1)$$

### 2.2 가용 대역폭 측정 기법

적응적 스트리밍 기술에서 가용 대역폭의 측정 기법은 비트율 선택 기법에게 정확한 가용 네트워크 대역폭을 측정하여 전달해주어야 한다. 기존의 제안된 측정 기법으로는 비디오 세그먼트 단위로 가용 대역폭을 측정하는 기법<sup>[8]</sup>, 매 round trip time (RTT)마다 가용 네트워크 대역폭의 샘플을 측정하고, exponential moving average를 통해 평균값을 계산하여 최종 가용 네트워크 측정값을 계산하는 기법<sup>[7]</sup>이 있다. 그러나 두 기법 모두 적응적 스트리밍에 적용하는 데 있어 다음과 같은 문제점이 있다. 세그먼트의 길이가 일반적으로 2-4 초임을 감안하면, RTT 기반 측정 기법은 상대적으로 매우 짧은 시간 동안의 가용 대역폭을 측정한다. 따라서, 측정된 RTT값이 다음 세그먼트를 다운로드 할 동안의 가용 네트워크 대역폭에 대한 예측값으로 사용되기에는 적합하지 않다. 세그먼트 기반 측정 기법의 경우에는, 매 세그먼트의 다운로드가 끝날 때 마다 측정값을 제공하여 앞서 언급한 문제를 완화

시킬 수 있다. 하지만, 네트워크 상황이 악화되어 측정 주기가 길어질 경우 (즉, 세그먼트 다운로드 시간이 길어질 경우), 가용 네트워크 대역폭의 현재 상태를 효율적으로 반영하지 못하는 상황이 발생하게 된다.

제안하는 비트율 적응 알고리즘인 QAAD에서는 두 기법의 장점을 수용하여 고정 주기로 가용 네트워크 대역폭을 샘플링 하고, exponential moving average를 통해 계산된 평균값을 가용 네트워크 대역폭의 측정값으로 하는 가용 네트워크 대역폭 측정 기법을 고안하였다. 가용 네트워크 대역폭의 샘플 측정값  $bw_{sample}$ 은  $\theta$  초 마다  $bw_{sample} = K/\theta$ 로 측정된다. 이 때  $K$ 는  $\theta$  초 동안 다운로드 된 데이터의 양 (단위: kilo byte)을 나타낸다. 그러면, 가용 네트워크 대역폭의 측정값  $bw_{estimated}$ 는 다음과 같이 계산된다.

$$bw_{estimated} = \omega * bw_{estimated} + (1 - \omega) * bw_{sample}. \quad (2)$$

이 때,  $\omega$ 는 exponential moving average의 가중치 파라미터로  $0 \leq \omega \leq 1$ 의 범위를 갖는다.

그림 2는 앞서 언급한 기존의 RTT 기반 및 세그먼트 기반 가용 대역폭 측정 기법과 제안하는 가용 대역폭 측정 기법을 비교한 결과를 나타낸다. 서로 동등한 가용 대역폭 환경에서 각 기법을 비교하기 위하여, 가용 대역폭은 Dummysnet<sup>[16]</sup>을 이용하여 다음과 같이 제한되었다. 13 초에서 3000 Kbps에서 1500 Kbps로, 23 초에서 1500 Kbps에서 3000 Kbps로, 33 초에서 다시 3000 Kbps에서 1500 Kbps로 제한되었다. 비디오 세그먼트의 길이는 2 초로 하였으며, 제안하는 대역폭 제안 기법의 파라미터  $\theta$ 와  $\omega$ 는 경험적으로, 각

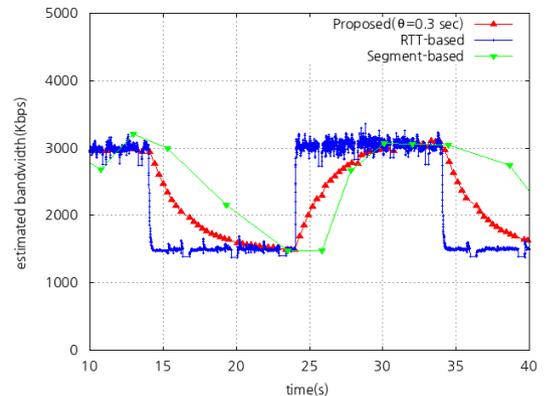


그림 2. 대역폭 측정 기법 비교  
Fig. 2. Comparison of the bandwidth estimation schemes

각 0.3 초, 0.85로 결정하였다. 먼저 RTT 기반의 측정 방식은 대역폭 변화에 매우 민감하게 반응하는 것을 확인할 수 있고, 따라서 멀티미디어 스트리밍에 적합하지 않다. 한편, 세그먼트 기반의 측정 방식은 너무 긴 시간 동안의 평균값을 취하기 때문에 가용 대역폭의 변화 추세를 빠르게 반영하지 못한다. 제안하는 주기적인 측정 기법은 적절한 주기를 통해 측정함으로써 앞서 언급한 두 기법을 잘 조화시켜 비트율을 선택해야 할 시점에 적합한 가용 대역폭 측정값을 제공하는 것을 확인할 수 있다.

### 2.3 비트율 선택 기법

QAAD의 비트율 선택 기법의 핵심 아이디어는 사용자가 느끼는 비디오 품질의 급격한 저하 (즉, 높은 비트율로 인코딩된 비디오를 시청하다가 갑자기 낮은 비트율로 인코딩된 비디오를 시청하는 경우)를 느끼지 않도록 하는 것이다. 이를 위해, 비트율을 선택할 때, 재생 버퍼 상태와 가용 네트워크 대역폭을 동시에 고려한다. 그림 3은 제안하는 비트율 선택 기법을 순서도를 통해 설명하는 그림이다. 먼저 이전 세그먼트의 다운로드가 끝나면, 이전에 요청된 비디오 품질 레벨  $l_{prev}$ 와 현재 시점에서 가용 대역폭을 넘지 않는 최대 품질 레벨인  $l_{best}$ 와의 차이를 비교하여 각각의 경우에 따라 다음 요청 품질 레벨  $l_{next}$ 를 다르게 결정한다.

구체적으로 1)  $l_{best} == l_{prev}$ , 2)  $l_{best} > l_{prev}$  그리고 3)  $l_{best} < l_{prev}$ 의 경우를 고려한다. 첫 번째의 경우 (즉,  $l_{best} == l_{prev}$ ), 현재의 가용 네트워크 대역폭이 지원할 수 있는 최대 품질 레벨이 이전에 요청한 품질 레벨과 같으므로, 이전 품질 레벨을 그대로 요청한다

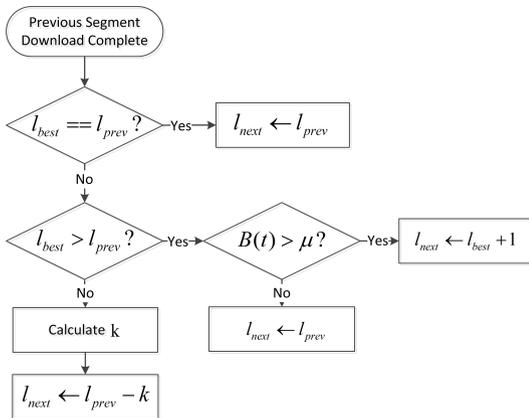


그림 3. QAAD의 비트율 선택 기법  
Fig. 3. Bitrate selection scheme in QAAD

(즉,  $l_{next} \leftarrow l_{prev}$ ). 이 경우에는, 가용 네트워크 대역폭보다 작은 비트율 중 가장 높은 비트율을 요청하게 되므로, 재생 버퍼의 변화가 거의 없다고 판단할 수 있다. 두 번째의 경우 (즉,  $l_{best} > l_{prev}$ ), 이전 요청 품질 레벨보다 현재의 가용 네트워크 대역폭이 지원할 수 있는 최대 품질 레벨이 더 높다. 이 때, 재생 버퍼 레벨을 고려하지 않고 이전 요청 품질 레벨 대비 더 높은 품질 레벨을 요청할 경우, 예기치 못한 가용 네트워크 대역폭 감소에 대비할 여분의 재생 버퍼량을 확보할 수 없다. 따라서 그 시점의 재생 버퍼량을 고려하여 다음 요청 품질 레벨을 결정한다. 즉, 현재 재생 버퍼 레벨  $B(t)$ 가 여분 재생 버퍼 보존 레벨  $\mu$ 보다 클 경우에만 다음 요청 품질 레벨을 1 만큼 증가시킨다 (즉,  $l_{next} \leftarrow l_{prev} + 1$ ). 세 번째의 경우 (즉,  $l_{best} < l_{prev}$ ), 현재 네트워크가 지원할 수 있는 최대 품질 레벨이 이전 요청 품질 레벨보다 더 낮으므로, 요청 품질 레벨을 이전 요청 레벨인  $l_{prev}$ 에서 감소시켜야 한다. 하지만, 재생 품질을 급격히 감소시킬 경우 (즉, 2 레벨 이상의 감소), 사용자가 재생 품질 저하를 느끼게 되며, 이러한 상황이 반복될 경우 사용자의 체감 품질을 급격히 저하시킨다. 따라서 이를 방지하기 위해 제안하는 비트율 선택 기법은 현재 시점에 저장되어 있는 비디오 데이터 (즉, 재생 버퍼)를 소비하여 최대한  $l_{prev}$ 에 가까운 재생 품질 레벨을 요청한다. 이 때, 예기치 못한 가용 네트워크 대역폭 감소로 인한 재생 버퍼 고갈을 방지하기 위해, 최소 재생 버퍼 보존 레벨  $\sigma$ 은 항상 보존하도록 한다. 따라서 활용 가능한 재생 버퍼량은  $B(t) - \sigma$ 이 된다. 이 때,  $l_{prev} - k$  ( $k = 0, 1, 2, \dots$ )를 소비 가능한 재생 버퍼량  $B(t) - \sigma$ 으로 다운받을 수 있는 최대 재생 품질 레벨로 정의한다. 이 때,  $B(t) - \sigma$ 를 소비하여 다운받을 수 있는 재생 품질  $l_{prev} - k$ 의 세그먼트 개수  $n_{l_{prev} - k}$ 는 다음과 같이 계산된다.

$$n_{l_{prev} - k} = \left\lceil \frac{B(t) - \sigma}{\tau(b(l_{prev} - k)/bw_{estimated} - 1)} \right\rceil. \quad (3)$$

이 때,  $\lceil x \rceil$ 는  $x$ 에 대한 올림 함수이다. 분모는 현재 가용 네트워크 대역폭이  $bw_{estimated}$ 일 때, 재생 품질 레벨  $l_{prev} - k$ 의 세그먼트 하나를 다운로드 받는데 걸리는 시간이다. 따라서 소비 가능한 재생 버퍼량  $B(t) - \sigma$ 을 이로 나누고 올림 함수를 취하면 다운받을 수 있는 재생 품질  $l_{prev} - k$ 의 세그먼트 개수

$n_{l_{prev}-k}$ 가 되는 것이다. 결과적으로 최적의  $k$ 는  $n_{l_{prev}-k}$ 를 1보다 크거나 같게 하는  $k$  중 가장 큰 값으로 계산된다.

### III. 실험 결과

제안하는 DASH에서의 비트율 적응 알고리즘인 QAAD에 대한 성능을 실험적으로 평가하기 위해 DASH 테스트베드를 구축하였다. DASH 클라이언트는 DASH VLC Plugin<sup>[13]</sup>의 소스코드를 수정하여 구현하였고 DASH 서버는 Apache HTTP 서버<sup>[14]</sup>를 통해 구현하였다. DASH 표준에 부합하는 비디오 데이터를 생성하기 위해, DASHencoder<sup>[13]</sup>을 사용하여 비디오 영상 Big Buck Bunny Movie<sup>[15]</sup>를 8개의 비디오 비트율 (즉, 400 Kbps, 500 Kbps, 600 Kbps, 800 Kbps, 1000 Kbps, 1200 Kbps, 1600 Kbps, 그리고 2000 Kbps)로 인코딩하였고 각 버전을 2초 단위의 세그먼트로 분할하였다. 또한 가용 네트워크 대역폭을 임의로 제한하기 위하여 DASH 클라이언트에서 Dummynet<sup>[16]</sup>을 사용하였다. Dummynet은 호스트의 특정 세션에서 들어오는 패킷들을 가로챌으로써 사용자가 선언한 네트워크 대역폭, 네트워크 지연 시간 등을 에뮬레이션 해 주는 도구이다. 제안하는 가용 네트워크 대역폭 측정 기법의 파라미터인  $\theta$ 와  $\omega$ 는 각각 0.3 초 및 0.875로 설정하였다. 또한, 여분 재생 버퍼 보존 레벨  $\mu$  및 최소 재생 버퍼 보존 레벨  $\sigma$ 은 각각 10 초 그리고 3 초로 결정하였다.

제안하는 비트율 적응 알고리즘과의 성능 비교를 목적으로, 재생 버퍼를 고려한 기존의 비트율 적응 알고리즘인 QDASH<sup>[9]</sup>을 구현하였다. QDASH에서는 RTT 기반의 가용 대역폭 측정 기법을 사용하며, 그림

4의 비트율 선택기법을 사용한다. 현재 가용 네트워크 대역폭이 지원하는 최대 품질 레벨  $l_{best}$ 가 이전 요청 품질 레벨  $l_{prev}$ 보다 크거나 같을 경우 (즉,  $l_{best} \geq l_{prev}$ ),  $l_{best}$ 의 세그먼트를 요청한다 (즉,  $l_{next} \leftarrow l_{best}$ ). 이와 같이 급진적으로 요청 비트율을 증가시키게 되면, 예기치 못한 대역폭 감소에 대한 완충 작용을 할 재생 버퍼량이 보존되지 않는다. 또한 가용 대역폭이 변동함에 따라  $l_{best}$  또한 그에 맞게 변동할 것이므로, 이러한 방식은 비디오 품질이 향상되었다가 감소되는 반복적인 상황을 초래, 사용자의 체감 품질을 저하시키게 된다. 한편, 만약  $l_{best}$ 가 이전 요청 품질 레벨  $l_{prev}$ 보다 두 단계 이상 작을 경우 (즉,  $l_{best} < l_{prev} - 1$ ), QDASH에서는 품질 레벨의 차이가 크다고 판단하여 재생 버퍼 레벨을 고려하여  $l_{best} + 1$ 의 세그먼트의 다운로드가 가능하다면 (즉,  $n_{l_{best}+1} \geq 1$ ),  $l_{best} + 1$ 의 세그먼트를 요청하고 (즉,  $l_{next} \leftarrow l_{best} + 1$ ), 재생 버퍼 레벨이 부족하여 불가능하다면  $l_{best}$ 로 요청한다 (즉,  $l_{next} \leftarrow l_{best}$ ). 하지만 QDASH에서는  $n_{l_{best}+1}$ 를 계산할 때, 최소 버퍼 보존 레벨을 고려하지 않는다 (즉,  $\sigma = 0$ ). 따라서 예기치 못한 가용 네트워크 대역폭 감소에 대해 완충작용을 할 최소 버퍼량이 확보되지 않으므로, 재생 끊김 등이 발생할 수 있다. 만약  $l_{best}$ 가 이전 요청 품질 레벨  $l_{prev}$ 보다 한 단계만 작을 경우, QDASH에서는 심각한 품질 저하가 아니라고 판단,  $l_{best}$ 의 세그먼트를 요청한다 (즉,  $l_{next} \leftarrow l_{best}$ ). 두 기법의 성능 비교를 위해, 두 종류의 가용 네트워크 대역폭 환경에 대한 실험을 수행하였다. 구체적으로 1) Step-down 실험 그리고 2) Fluctuation 실험을 수행하였다. 첫 번째 실험에서는, 가용 네트워크 대역폭을 30 초쯤에 2200 Kbps에서 800 Kbps로 변경하였다. 한편 두 번째 실험에서는, 가용 네트워크 대역폭을 매 4 초마다 2100 Kbps에서 800 Kbps로, 800 Kbps에서 2100 Kbps로 변경하였다. 그림 5는 Step-down 실험에 대한 각 기법의 시간에 따른 요청 비트율 그래프 및 재생 버퍼 레벨 그래프를 나타낸다. 초기에 QDASH는 재생 버퍼 레벨을 고려하지 않고 비트율을 결정하므로 QAAD보다 높은 비트율을 요청한다. 한편 QAAD는 여분 재생 버퍼 보존 레벨인 10 초의 비디오 데이터가 버퍼에 저장 될 때까지 비트율을 높이지 않는 것을 확인할 수 있다. 결과적으로 30 초 이후, QDASH에서는 재생 버퍼가 보존되지 않아, 급격하게 비트율을 감소시키며, 사용

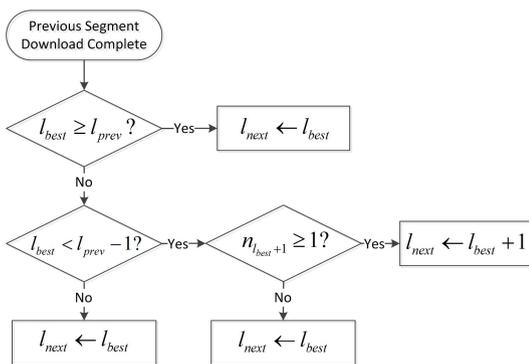


그림 4. QDASH의 비트율 선택 기법  
Fig. 4. Bitrate selection scheme in QDASH

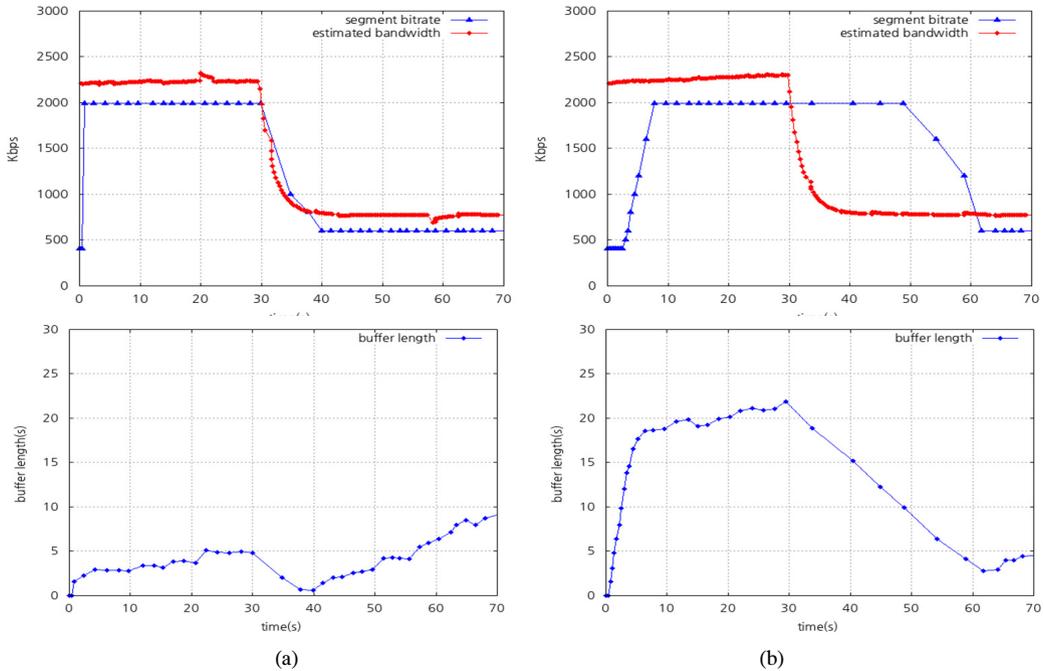


그림 5. Step-down 실험 ((a)QDASH 그리고 (b)QAAD)  
 Fig. 5. Step-down test ((a)QDASH and (b)QAAD)

자로 하여금 재생 품질의 급격한 저하를 느끼게 한다. 또한 가용 네트워크 대역폭보다 낮은 비트율을 요청할 때까지의 시간을 비교해 보면, QDASH는 40 초, QAAD는 63 초이다. 이는 가용 네트워크 대역폭이 감소한 상황에서도 QAAD가 QDASH보다 약 23 초의 추가적인 시간 동안 사용자들로 하여금 비디오 품질 저하를 느끼지 않게 함으로써, 사용자 체감 품질을 향상시켰음을 의미한다.

그림 6은 Fluctuation 실험에 대한 각 기법의 시간에 따른 요청 비트율 그래프 및 재생 버퍼 레벨 그래프를 나타낸다. 요청 비트율 그래프를 보면, QDASH는 가용 네트워크 대역폭에 매우 민감하게 반응하는 것을 알 수 있다. 즉, 가용 네트워크 대역폭이 증가하는 상황에서는 급격히 요청 비트율을 증가시키며, 가용 네트워크 대역폭이 감소하였을 때에는 급격하게 요청 비트율을 감소시키는 것이다. 이로 인해, QDASH의 재생 버퍼는 보존이 되지 않음을 확인할 수 있다. 한편, QAAD는 가용 네트워크 대역폭이 매우 높아도 재생 버퍼를 고려하며 비트율을 증가시키고 따라서 재생 버퍼 레벨을 보존하는 것을 확인할 수 있다. 결과적으로, 가용 네트워크 대역폭이 매우 급변하는 상황에서도 QAAD는 안정적으로 1600 Kbps의 비트율을 요청하며, 비트율의 변경 (품질 레벨의 변경)

횟수를 최소화 할 수 있다. 한편, QDASH는 예기치 못한 가용 네트워크 대역폭에 대처할 만한 재생 버퍼량이 존재하지 않으므로, 16 초에 재생 버퍼의 고갈로 인한 재생 끊김 현상이 발생한다. 반면 QAAD는 재생 버퍼를 보존하는 특성으로 인해 실험 도중 단 한 번도 재생 끊김 현상이 발생하지 않았다.

#### IV. 실험 결과

본 고에서는 DASH 기반의 적응적 멀티미디어 스트리밍 기술의 비디오 비트율 적응 알고리즘을 제안하였다. 제안하는 알고리즘인 QAAD는 재생 끊김을 방지하고 비디오 품질 변화의 횟수를 최소화 하여 사용자의 체감 품질을 향상시킨다. DASH 테스트베드 상에서 실험을 수행한 결과, 매우 가변적인 네트워크 환경에서도 QAAD가 기존의 기법보다 더 안정적인 비디오 화질을 제공하는 것을 확인하였다. 향후에는 연결이 간헐적인 Delay Tolerant Networks (DTNs)에서의 적응적 스트리밍 기법에 대한 연구를 진행할 예정이다.

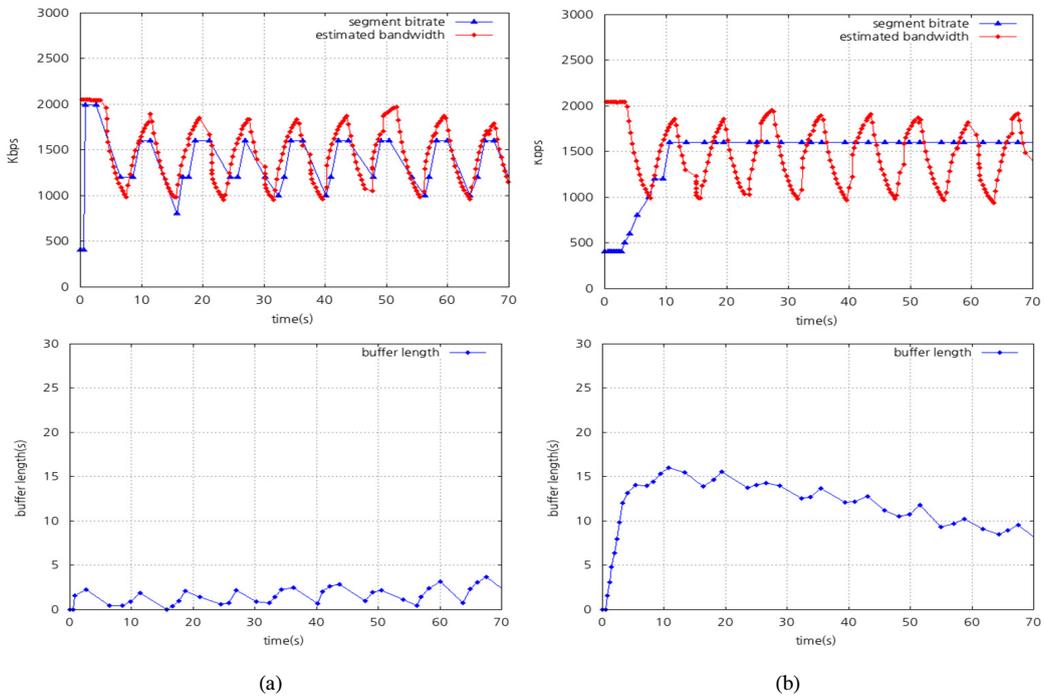


그림 6. Fluctuation 실험.((a)QDASH 그리고 (b)QAAD)  
 Fig. 6. Fluctuation test ((a)QDASH and (b)QAAD)

### References

[1] D. Suh, I. Jang, and S. Pack, "QoE-enhanced Adaptation Algorithm over DASH for Multimedia Streaming," in *Proc. ICOIN 2014*, Feb. 2014.

[2] ISO/IEC IS 23009-1: "Information technology - Dynamic adaptive streaming over HTTP (DASH) - Part 1: Media presentation description and segment formats," 2012.

[3] S. Akhshabi, A. Begen, and C. Dovrolis, "An experimental evaluation of rate-adaptation algorithms in adaptive streaming over http," in *Proc. ACM MMSys 2011*, Feb. 2011.

[4] Y. Xu, Y. Zhou, and D.-M. Chiu, "Analytical QoE models for bit-rate switching in dynamic adaptive streaming systems," *IEEE Trans. Mobile Comput.*, pp. 1-15, 2013.

[5] D. Jarnikov, "Client intelligence for adaptive streaming solutions," in *Proc. IEEE ICME 2010*, Jul. 2010.

[6] T. Thang, Q. Ho, J. Kang, and A. Pham, "Adaptive streaming of audiovisual content using MPEG DASH," *IEEE Trans. Consumer Electron.*, vol. 58, no. 1, pp. 78-85, Feb. 2012.

[7] Y. Kim, J. Shin, and J. Park, "Design and implementation of a network-adaptive mechanism for HTTP video streaming," *ETRI J.*, vol. 35, no. 1, pp. 27-34, Feb. 2013.

[8] C. Liu, I. Bouazizi, and M. Gabbouj, "Rate adaptation for adaptive HTTP streaming," in *Proc. ACM MMSys 2011*, Feb. 2011.

[9] R. Mok, X. Luo, E. Chan, and R. Chang, "QDASH: A QoE-aware DASH system," in *Proc. ACM MMSys 2012*, Feb. 2012.

[10] N. Kim, T. Song, W. Kim, and S. Pack, "A scalable video coding (SVC)-Aware retransmission scheme for multimedia streaming in IEEE 802.11 WLANs," *J. KICS*, vol. 39, no. 2, pp. 95-101, Feb. 2014.

[11] H. Joo, D. Lee, S. Han, and H. Song, "Efficient video streaming system over heterogeneous wireless networks," *J. KICS*, vol. 37, no. 2, pp. 110-119, Feb. 2012.

- [12] K. Choi, J. Yoo, and J. Kim, "Monitoring-based coordination of network-adaptive FEC for wireless multi-hop video streaming," *J. KICS*, vol. 36, no. 2, pp. 114-126, Feb. 2011.
- [13] C. Müller and C. Timmerer, "A VLC media player plugin enabling dynamic adaptive streaming over HTTP," in *Proc. ACM MM 2011*, Nov. 2011.
- [14] The Apache Software Foundation. The Apache HTTP server. <http://www.apache.org/>
- [15] Big Buck Bunny Movie. <http://www.bigbuckbunny.org/>
- [16] M. Carbone and L. Rizzo, "Dummynet revisited," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 2, pp. 12-20, Mar. 2010.

**서 동 은 (Dongeun Suh)**



2012년 2월 : 고려대학교 전기  
전자전파공학부 졸업  
20012년 9월~현재 : 고려대학교  
전기전자공학과 석박사 통합  
과정  
<관심분야> 무선 이동 네트워  
크, 멀티미디어 스트리밍

**장 인 선 (Insun Jang)**



2011년 2월 : 고려대학교 전기  
전자전파공학부 졸업  
2011년 2월~현재 : 고려대학교  
전기전자공학과 석박사 통합  
과정  
<관심분야> 소프트웨어 정의  
네트워킹, 다중 무선 인터페  
이스 네트워크

**백 상 현 (Sangheon Pack)**



2000년 2월 : 서울대학교 컴퓨  
터공학부 졸업  
2005년 2월 : 서울대학교 전기  
컴퓨터공학부 박사졸업  
2007년 3월~현재 : 고려대학교  
전기전자전파공학부 부교수  
<관심분야> 미래인터넷, 무선  
이동 네트워크