논문 2022-17-36

A3C 기반의 강화학습을 사용한 DASH 시스템
(A DASH System Using the A3C-based Deep Reinforcement Learning)

최 민 제, 임 경 식*
(Minje Choi, Kyungshik Lim)

Abstract : The simple procedural segment selection algorithm commonly used in Dynamic Adaptive Streaming over HTTP (DASH) reveals severe weakness to provide high-quality streaming services in the integrated mobile networks of various wired and wireless links. A major issue could be how to properly cope with dynamically changing underlying network conditions. The key to meet it should be to make the segment selection algorithm much more adaptive to fluctuation of network traffics. This paper presents a system architecture that replaces the existing procedural segment selection algorithm with a deep reinforcement learning algorithm based on the Asynchronous Advantage Actor-Critic (A3C). The distributed A3C-based deep learning server is designed and implemented to allow multiple clients in different network conditions to stream videos simultaneously, collect learning data quickly, and learn asynchronously, resulting in greatly improved learning speed as the number of video clients increases. The performance analysis shows that the proposed algorithm outperforms both the conventional DASH algorithm and the Deep Q-Network algorithm in terms of the user's quality of experience and the speed of deep learning.

Keywords : DASH, Deep Reinforcement Learning, A3C, Adaptive Video Streaming

Ⅰ. 서 론

최근 모바일 네트워크 환경에서 고품질 스트리밍 서비스를 제공하기 위하여 적응형 비디오 스트리밍 (adaptive video streaming) 기술이 널리 사용되고 있다. DASH (Dynamic Adaptive Streaming over HTTP) [1, 2]는 그 중 대표적인 기술로서 단말은 현재 환경에서 끊김 없는 서비스를 제공하는데 적합한 비트율 (bit rate)을 계산하고 이에 해당하는 크기를 갖는 세그먼트 (segment)를 서버로부터 내려받아 서비스를 제공한다. 이때 서버는 제공할 비디오 콘텐츠를 미리 정의된 다양한 비트율로 인코딩하여 다양한 품질을 갖는 일련의 세그먼트 형태로 저장하고 이에 대한 정보를 Media Presentation Description (MPD)에 담아 서비스가 시작할 때 단말에 전달한다.

이 기술의 핵심은 현재 사용하고 있는 네트워크의 가용 대역폭을 단말이 실시간으로 계산하고 다음에 가져올 비디오 세그먼트의 비트율을 선택하기 위하여 DASH Adaptive BitRate (ABR) 알고리즘을 사용하는 데 있다. 이 알고리즘은 처리량 (throughput), 지연시간 (latency), 버퍼길이 (buffer length), 프레임 드롭 횟수 (dropped frames)를 이용하여 네트워크 환경을 분석하고 간단한 절차적 알고리즘을 통해 적절한 품질의 세그먼트를 선택한다. 그러나 이러한 단순하고 사전에 정의된 절차적 알고리즘만으로는 급변하는 네트워크 환경에 실시간으로 대처하기에는 상당한 어려움이 있으므로 사용자의 QoE (quality of experience)를 개선할 수 있는 다양한 네트워크 성능 메트릭 (performance metrics)에 대하여 많은 연구가 진행되어왔다 [3].

한편으로 최근에는 유무선 복합 네트워크 환경에서의 급변하는 유무선 선로 특성과 단말의 다양한 이동성을 고려하여 적응형 비디오 스트리밍에도 딥러닝 기술을 접목하는 연구가 진행되고 있다. 이러한 연구는 DASH의 비트 처리율에 기반한 절차적 알고리즘을 심층 강화학습 (deep reinforcement learning) 알고리즘으로 대체하여 사전에 절차적으로 정의할 수 없는 다양한 네트워크 상황이 발생할 때 적응적으로 더 높은 대역폭을 활용하면서도 끊김 없는 고품질 서비스를 제공하는 것을 목표로 하고 있다 [4, 5]. 이 연구에서는 기존 DASH ABR 알고리즘은 손실률이 낮은 네트워크 환경에서도 잦은 품질 변화를 초래하여 평균 품질을 하락시키지만, Deep Q-Network (DQN) 기반 알고리즘은 신속하게 적합한 품질을 선택하여 품질 변화 횟수를 줄이고 버퍼 길이를 높게 유지하여 안정적으로 품질을 유지함을 보인다. 더구나 손실률이 매우 높은 네트워크 환경에서의 DASH ABR 알고리즘은 네트워크에서 수용하지 못하는 품질을 선택함에 따라 버퍼가 고갈되어 리버퍼링 (rebuffering) 횟수를 높이고 리버퍼링 지속시간을 길게 하여 QoE를 저하시키지만, DQN 기반 알고리즘은 현재의 상

*Corresponding Author (limkyungshik@gmail.com)
Received: Jun. 26, 2022, Revised: Aug. 3, 2022, Accepted: Sep. 16, 2022.
Minje Choi: Kyungpook National University (MS)
Kyungshik Lim: Kyungpook National University (Prof.)

ⓒ IEMEK J. Embed. Sys. Appl. 2022 Oct. 17(5) 297-307
ISSN : 1975-5066
http://dx.doi.org/10.14372/IEMEK.2022.17.5.297

태에서 수용 가능한 범위의 적합한 품질을 선택하여 버퍼 길이를 적절히 유지함으로써 리버퍼링 횟수와 리버퍼링 지속시간을 줄여 안정적인 비디오 스트리밍 서비스를 제공할 수 있음을 보인다.

그러나 이러한 DQN 기반 알고리즘이 다양한 상황에서 최적의 성능을 내기 위해서는 시스템이 많은 학습데이터를 수집하고 학습모델을 생성해야 하므로 비교적 많은 메모리와 시간이 소요된다는 단점이 있다. 또한 각 단말이 처한 다양한 네트워크 상황뿐만 아니라 다양한 단말의 특성들이 모두 반영될 수 있는 학습 모델이 필요하다. 이러한 요구사항들을 반영하기 위하여 본 논문에서는 다수의 클라이언트에서 비디오 스트리밍을 동시에 비동기적으로 학습할 수 있는 Asynchronous Advantage Actor-Critic (A3C) [6, 7] 모델을 적용하여 DASH 시스템을 개발한다. 제안된 A3C 기반 알고리즘은 단말의 수가 증가함에 따라 DQN 기반 알고리즘에 비해 학습속도가 빠르게 개선되고 다양한 네트워크와 단말 특성에 적응적으로 고품질 비디오를 선택할 수 있음을 보인다. 본 논문의 2장에서는 배경 기술들에 대해 살펴보고, 3장에서는 제안하는 모델에 대하여 설명한 후, 4장에서는 실험 환경 및 실험 평가를 하고 5장에서 결론을 맺는다.

Ⅱ. 관련 연구

1. Dynamic Adaptive Streaming over HTTP (DASH)

DASH는 현재 네트워크 상태에 맞춰 비디오 품질을 조정함으로써 네트워크 상태의 변화에 따라 생길 수 있는 비디오 끊김 현상을 개선하기 위해 채택된 기술이다. 고전적인 HTTP 비디오 스트리밍과 비교하여 DASH는 간헐적으로 발생하는 비디오 중단 횟수를 줄이고 더 높은 대역폭을 활용하여 고품질 스트리밍을 가능하게 한다 [3]. 이는 일반적으로 사용자에게 더 높은 QoE를 제공하는데 [8], 이 기술의 핵심은 클라이언트 측에서 네트워크 및 단말기의 환경에 적합한 비디오 품질을 선택하는 것이다.

그림 1에서 보는 바와 같이 DASH는 비디오 세그먼트를 제공하는 서버와 비디오를 재생하는 클라이언트로 구성되어 있다. 서버는 비디오 콘텐츠를 해상도 (resolution), 비트율, 초당 프레임 (frame) 수, 압축률 등 다양한 기준으로 품질을 나누고 세그먼트 단위로 인코딩된 비디오를 저장한다. 세그먼트의 크기는 네트워크 환경의 변화에 빠르게 반응할 수 있을 만큼 충분히 작도록 설정되고 이에 대한 정보는 MPD에 저장된다. 클라이언트는 서버로부터 내려받은 MPD에 있는 비디오 정보를 바탕으로 자신의 네트워크 환경에 적합한 비디오 품질을 결정하고 해당 세그먼트를 서버에 요청한다. 여기서 비디오 품질을 선택하는 DASH ABR 알고리즘은 사용자의 QoE를 최대한 향상시킬 수 있어야 한다. 가장 일반적인 방식은 네트워크 상태를 처리량, 지연시간, 버퍼길이,

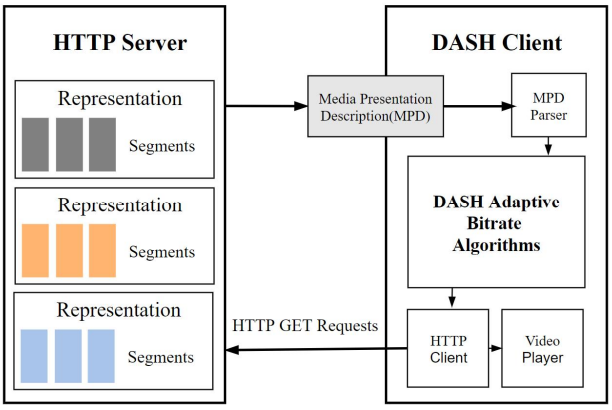

그림 1. DASH 시스템 구조
Fig. 1. The DASH System Architecture

프레임 드롭 횟수로 표현하고 이들을 기반으로 간단한 네 가지 절차적 규칙을 통해 품질을 선택한다. 이러한 품질 선택 알고리즘은 DASH의 성능을 결정하는 핵심 부분이기 때문에 표준화에서 제외된 개방 모듈로서 정의하고 이를 개선하려는 많은 연구가 진행되고 있다 [3].

2. 강화학습 (Reinforcement Learning)

강화학습은 일련의 행동 (action)을 하고 보상 (reward)을 받는 과정을 통해 미래에 수행할 행동에 대한 최적의 정책 (policy)을 학습하는 방법이다. 강화학습은 현재 상태에서 행동을 연속적으로 선택하는 순차적인 문제로 정의되는데 이를 상태 (S), 행동 (A), 보상 (R), 상태 변환 확률 (P), (할인률)γ로 구성된 Markov Decision Process (MDP)로 정의할 수 있다. 상태 (S)는 에이전트 (agent)가 관찰 가능한 상태의 집합이고 행동 (A)은 에이전트가 임의의 상태에서 취할 수 있는 행동의 집합이다. 보상 (R)은 학습의 방향을 이끌어주는 정보로써 에이전트가 행동을 했을 때 받는 값이다. 그리고 상태 변환 확률 (P)은 특정 상태에서 행동을 취한 후 도달할 수 있는 상태에 대한 확률이다. 마지막으로 할인율 (γ)은 0과 1사이의 값으로 현재 상태에 가까운 보상이면 큰 가치를 가질 수 있도록 하며 미래에 받을 보상의 가치를 줄이는 역할을 한다. 강화학습은 높은 보상을 받을 수 있는 행동의 확률을 높임으로써 최적의 정책을 찾는다. 환경 (environment)은 에이전트가 취한 행동에 대한 보상을 알려준다. 에이전트와 환경은 상호작용하며 실제로 받은 보상을 가지고 가치함수 (value function)와 정책을 바꿔나간다. 강화학습은 앞으로 받을 보상에 대해 예측을 할 수 있도록 미래에 받을 보상의 총합을 나타내는 가치함수를 정의한다.

강화학습을 위한 방법은 여러 가지가 있는데 대표적으로 Deep Q-Network (DQN)이 있다 [9]. Q-Learning은 상태와 환경을 Q-table 형태로 표현하여 업데이트하는 방식으로 학습한다. 그러나 최근의 문제들은 상태 공간이 매우 커지며

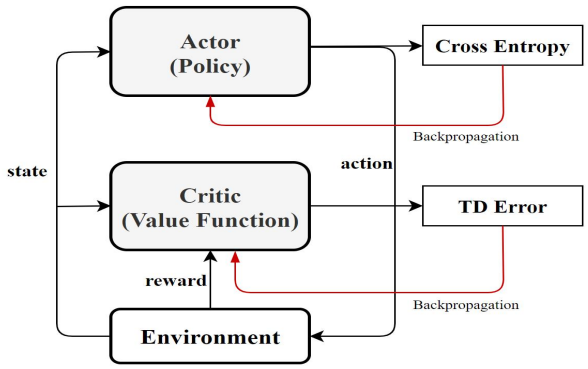

그림 2. Actor-Critic의 구조
Fig. 2. The Structure of Actor-Critic

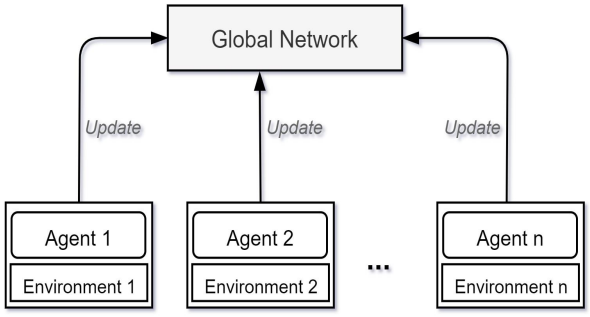

그림 3. A3C의 전역신경망 업데이트 방식
Fig. 3. A3C's Updating Method with The Global Neural Network

최적의 정책을 찾는데 많은 에피소드를 필요로 한다. DQN은 Q-Learning에 신경망을 도입하여 상태 공간이 큰 문제에서도 정책을 학습할 수 있도록 하였다.

3. Asynchronous Advantage Actor-Critic (A3C)

Actor-Critic [7]은 policy gradient 방법으로서 정책신경망 (actor)과 가치신경망 (critic)이라는 두 개의 인공신경망으로 파라미터에 대한 정책을 직접 학습하는 강화학습 알고리즘이다 (그림 2). 정책신경망은 정책을 근사하며 행동을 선택하는 역할을 하고 가치신경망은 가치함수를 근사하며 정책신경망이 선택한 행동에 대해 평가하는 역할을 한다. 에이전트 (agent)는 환경 (environment)으로부터 상태 (state)를 받아오고 이를 정책신경망과 가치신경망의 입력으로 넣는다. 그리고 정책신경망의 출력으로 행동 (action)을 선택하고 다른 행동을 따르는 것보다 얼마나 더 좋은 결정인지 판단하기 위하여 어드밴티지 (advantage)를 구해 신경망을 업데이트한다. 가치신경망은 환경으로부터 행동에 대한 보상을 받아 시간차 에러 (temporal difference error)를 구해 신경망을 업데이트한다.

A3C는 그림 3에서와 같이 하나의 전역신경망 (global network)과 동일한 구조를 갖는 다수의 에이전트로 구성되어 있으며, 이들 각각은 정책신경망과 가치신경망을 가지고 있다 [6, 7]. 각 에이전트는 독립된 환경과 상호작용하며 로컬신경망에 따라 행동을 선택하고 오류함수와 그레디언트를 계산하여 전역신경망을 업데이트한다. 그 후 업데이트된 전역신경망의 가중치를 로컬신경망의 가중치로 복사한다. 이는 서로 다른 단말의 특성과 네트워크 상황을 짧은 시간에 학습시켜 어떤 주어진 시점에서 네트워크 전체 상황을 고려한 최적의 세그먼트 품질을 선택해야 하는 경우에 매우 적합한 모델이라 하겠다.

Ⅲ. A3C 기반 DASH 시스템

DASH ABR 알고리즘을 Deep Q-Network (DQN) 알고리즘으로 대체한 기존 연구는 QoE를 개선할 수 있는 새로운 성능 메트릭을 도입하여 사전에 절차적 알고리즘으로 정의할 수 없는 다양한 네트워크 상황이 발생할 경우 적응적으로 더 높은 대역폭을 활용하면서도 끊김 없는 고품질 서비스를 제공하는 방법을 소개하고 있다 [4, 5]. 그러나 이러한 방법은 싱글 에이전트 학습 방식으로 비교적 많은 메모리와 시간이 소요된다는 단점이 있을 뿐만 아니라 다양한 단말이 처한 다양한 네트워크 상황을 반영하는 데 어려움이 있다. 따라서 본 논문에서는 DASH ABR 알고리즘을 분산 강화학습 알고리즘인 A3C 알고리즘으로 대체하여 이러한 단점을 극복하고자 한다.

1. 시스템 구조

그림 4는 비디오 서버, 클라이언트 그리고 A3C 서버로 구성된 A3C 기반 DASH 시스템의 전체 구조를 나타낸다. 비디오 서버는 다양한 비트율로 인코딩된 비디오 세그먼트를 준비하고 있어 클라이언트의 요청에 따라 선택된 품질의 세그먼트를 전송한다. 기본적으로 비디오를 재생하는 기능을 가지고 있는 클라이언트는 프록시 (proxy) 모듈을 통하여 현재의 네트워크 상태를 측정하여 이를 A3C 서버로 전송하고 결과로 받은 세그먼트 품질을 비디오 서버로 요청한다.

A3C 서버는 하나의 전역신경망과 각 클라이언트에 일대일로 대응하는 여러 개의 쓰레드 (thread)를 가지는데, 각 쓰레드는 해당 클라이언트로부터 네트워크 상태를 받아 비디오 품질을 반환하는 역할과 그 과정에서 나오는 데이터 샘플을 가지고 학습모델을 생성하는 역할을 수행한다. 쓰레드는 전역신경망과 동일한 구조를 가지는 로컬신경망을 가지고 있으며 대응된 클라이언트에 적합한 비디오 품질을 선택하여 전송하고 전역신경망을 업데이트한다. 이러한 시스템 구조는 각기 다른 네트워크 상황에 있는 여러 개의 단말을 수용하여 병렬로 스트리밍 서비스를 진행하면서 빠르게 학습데이터를 생성하고 하나의 학습모델로 업데이트할 수 있는 시스템을 가능하게 한다.

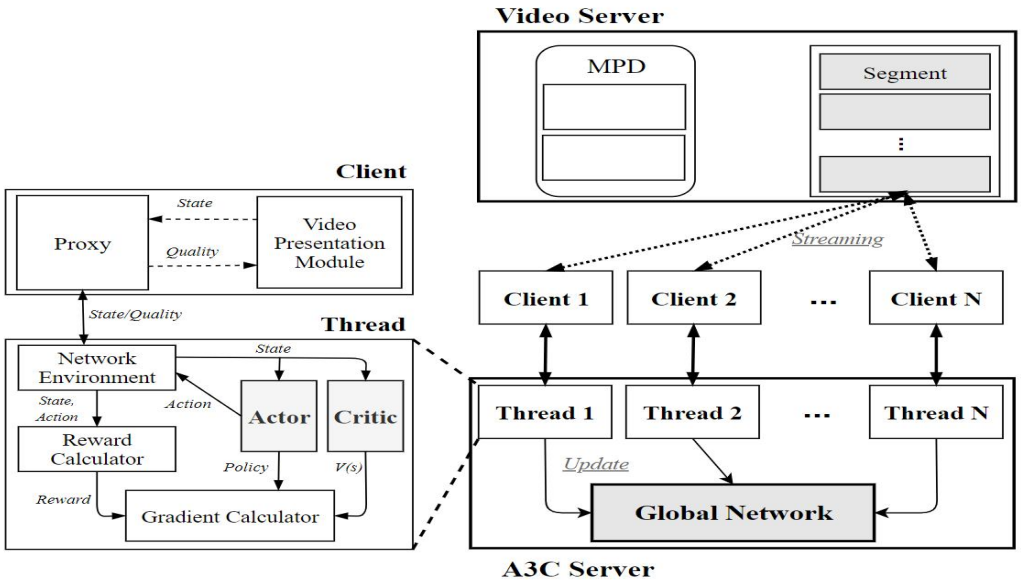

그림 4. A3C 기반 DASH 시스템의 구조
Fig. 4. The Architecture of The A3C-based DASH System

2. 보상 정의

A3C 기반 DASH 알고리즘에서는 비디오 품질 선택을 행동 (action)으로 정의하고 그 행동에 따른 네트워크 상태 변수들의 값을 가지고 보상을 정의했다. 보상의 계산에 사용되는 메트릭은 다양한 성능 메트릭 가운데 QoE에 가장 많은 영향을 미친다고 판단되는 평균 품질, 품질 변환 횟수, 버퍼 길이, 버퍼 길이의 변화, 버퍼 타겟으로 선정하였다. 선정된 각 메트릭의 가중치는 반복된 모든 조합의 실험 중에서 최고의 성능을 내는 경우의 가중치로 고정하였다.

$$
\begin{aligned}
R = {} & (0.2{*}r_q)+(0.1{*}r_{qswitch})+(0.2{*}r_b)+(0.3{*}r_{bswitch}) \\
& +(0.2{*}r_{btarget}), \quad where \\
& r_q = a_t, \\
& r_{qswitch} = a_{t+1} - a_t, \\
& r_b = \begin{cases} b_{a_{t+1}} & \text{if}(b_{a_{t+1}} > 0) \\ -30 & otherwise, \end{cases} \\
& r_{bswitch} = b_{a_{t+1}} - b_{a_t}, \text{ and} \\
& r_{qswitch} = a_{t+1} - a_t.
\end{aligned} \tag{1}
$$

식 (1)에서 r_q와 $r_{qswitch}$는 각각 t 단계에서 선택한 비디오 품질, 즉 행동 (a_t)과 t와 t+1 단계에서 선택한 비디오 품질의 차이를 반영한다. 즉, 높은 비디오 품질을 선택하거나 t 시점보다 t+1 시점에서 비디오 품질이 상승한다면 높은 보상을 받는다. $r_{bswitch}$는 행동 (a_t)을 했을 때, 즉 비디오 품질 (a_t)를 선택했을 때 변화하는 버퍼 길이를 나타내며 버퍼 길이가 증가하는 경우에 높은 보상을 받는다. 버퍼 타겟 ($r_{btartget}$)은 최상의 비디오 품질을 선택하여 품질의 변화가 발생하지 않는 경우 버퍼 타겟을 60으로 고정하여 버퍼에 많은 양의 비디오 데이터를 보유할 수 있도록 하는 변수이다. 그리고 r_b는 버퍼의 길이가 음수가 되었을 경우 즉, 재생 중단 현상 (rebuffering)이 발생했을 경우 보상에 -30 값을 주었고 아닐 경우 현재 버퍼 길이의 값을 주었다. 이렇게 사용하는 모든 메트릭에 대하여 보상변수들을 정의하고 각 보상변수들에 가중치를 적용하여 종합적으로 최종 보상 (R)을 계산한다. 가중치는 사용자의 QoE를 최대화할 수 있어야 결정되어야 하므로 가중치의 비율을 구하기 위해 많은 조합에 대하여 실험을 진행하였고 이를 통해 검증된 최적의 비율로 설정하였다.

3. 알고리즘 개요

하나의 클라이언트 (thread) 관점에서 A3C 기반 DASH 알고리즘의 동작 개요는 다음과 같다. 그림 4의 글로벌 네트워크는 Actor-Critic 기반으로 학습을 하며 정책신경망 (actor, θ)과 가치신경망 (critic, θ_v)을 가지고 각 쓰레드 또한 로컬 인공신경망 (θ', θ'_v)을 가진다. 쓰레드는 클라이언트로부터 네트워크 상태를 받아 자신의 정책신경망 (actor)으로 전달하고 비디오 품질을 반환받아 클라이언트로 전송한다. 이 과정을 $t_{\max}$번 반복한 후 보상값 (R)을 계산하고 네트워크를 업데이트하기 위한 그래디언트 ($d\theta$, $d\theta'$)를 구한다. 그리고 계산한 그래디언트로 글로벌 네트워크의 정책신경망과 및 가치신경망을 업데이트한다. 다수의 클라이언트 쓰레드들이 동시에 동작하며, 비동기적으로 글로벌 네트워크를 업데이트한다. 이처럼 각각 다른 환경에서 동작하는 클라이언트

로부터 학습데이터를 수집하게 되면 학습데이터 간의 높은 상관관계를 해결할 수 있으며 비동기적 업데이트를 통해 클라이언트 수에 비례하여 학습속도를 증가시킬 수 있다.

Ⅳ. 실험 및 성능 분석

1. 실험 환경

본 실험에서 학습의 효율성과 다양한 네트워크 환경을 설정하기 위해 network simulator (ns-3)를 이용하여 시뮬레이션하였다. ns-3 네트워크 환경 변수에는 대역폭 (bandwidth), 지연시간 (latency), 손실률 (error rate)이 있는데 LTE 모바일 환경을 가정하기 위해 대역폭을 40Mbps로 고정하고 지연시간과 손실률을 조합하여 다양한 네트워크 환경을 설정하고 실험을 진행하였다. 표 1은 실험에 사용된 네트워크 환경 변수와 설정값을 나타낸다. 실험에는 Jonathan Kua, Grenville Armitage 그리고 Philip Branch이 개발한 ns-3 DASH 시뮬레이션 모델 [10]을 기본적으로 사용하였으나, DASH ABR 알고리즘 부분을 A3C 서버에 케라스로 구현한 DQN과 A3C 기반 알고리즘으로 대체하였다. 실험에 사용된 비디오는 640초 분량이며 3 내지 4초 분량의 세그먼트로 분할하였다. 세그먼트는 화질과 비트율을 기준으로 품질을 0에서 9까지 10단계로 나누어 인코딩되어 서버에 미리 준비하였다. 품질 9의 세그먼트가 비트율이 가장 높고 화질이 좋으며 단계가 낮을수록 비트율과 화질이 감소한다. 표 2는 단계별 세그먼트에 대한 정보를 나타낸다.

실험에서 DQN은 클라이언트 수를 1개로 고정시켰고 A3C는 클라이언트 수를 1개에서 8개까지 증가시키며 다양한 네트워크 환경에서 학습을 진행했다. 두 알고리즘 간 학습속도의 차이를 나타내기 위해 비디오의 재생 시간에 따라 선택하는 비디오 품질과 보상의 변화를 측정하였다. 또한 ns-3 네트워크 환경 변수인 지연시간과 손실율의 다양한 변화에 따라 DASH, DQN, A3C 기반 알고리즘들이 각각 선택하는 세그먼트의 평균 품질과 리버퍼링 횟수를 비교하였다.

표 1. 실험에 사용된 네트워크 환경 변수
Table 1. Network Environments Parameters

Parameters	Value
Bandwidth	40Mbps
Latency	10ms, 30ms, 50ms, 100ms
Error rate	0.001, 0.01, 0.1

표 2. 실험에 사용된 세그먼트 정보
Table 2. Segment Information

Quality	Resolution	Value
0	320×180	200 kbps
1	320×180	400 kbps
2	480×270	600 kbps
3	640×360	800 kbps
4	640×360	1000 kbps
5	789×432	1500 kbps
6	1024×576	2500 kbps
7	1280×720	4000 kbps
8	1920×1080	8000 kbps

2. 성능 분석

2.1 학습 성능

그림 5와 그림 6은 비디오 플레이 시간에 따라 변화하는 평균 세그먼트 품질과 평균 보상을 각각 나타낸다. 그림에서 표시된 A3C의 첨자는 A3C 기반 DASH 알고리즘에서 비동기적으로 동시에 수행되는 클라이언트의 수를 가리킨다. 설정된 환경은 손실률이 0.001로 다른 실험 환경에 비해 낮으며 지연시간도 10ms로 가장 낮다. 이 환경에서는 클라이언트가 높은 품질의 세그먼트를 받아 재생할 수 있으므로 가능한 빠른 속도로 높은 비디오 품질을 선택할 수 있어야 한다. 학습 초기에는 DQN과 A3C 알고리즘 둘 다 네트워크 상태에 적응하지 못하고 낮은 품질의 세그먼트를 선택하는 경향을 보이고 있다. 학습 초기에는 평균 4와 5 사이의 품질을 갖는 세그먼트를 선택했고 이 선택에 따른 평균 보상의 값도 낮은 것을 볼 수 있다. 그러나 학습시간이 경과 할수록 평균 보상과 평균 세그먼트 품질이 증가하는 것을 볼 수 있다. 이는 클라이언트 (에이전트)가 비디오를 재생하면서 학습데이터를 수집하여 학습모델을 지속적으로 업데이트하고 있는 것을 의미한다.

DQN 알고리즘은 이 환경에서 학습이 수렴되어 안정적으로 평균 8과 9의 품질을 선택하려면 비디오 재생시간은 1,200분 정도이며 학습데이터는 19,200개 정도를 수집해야 한다. 즉, DQN 알고리즘은 처리량 40Mbps, 손실률 0.001, 지연시간 10ms 환경에서 비디오를 120번 정도 재생해야 주어진 네트워크 환경에 적합한 세그먼트 품질을 선택하는 학습모델을 생성할 수 있다는 것을 의미한다.

이에 반하여 클라이언트의 수가 1개인 A3C 알고리즘은 학습데이터 간의 연관성을 잘 해결하지 못하는 모습을 보이며 비디오 재생시간이 2,000분 이상이 지나도록 학습이 수렴되지 않는다. 그러나 클라이언트 수가 2개, 4개, 8개로 증가하면 그에 비례하여 학습속도가 빨라지는 것을 볼 수 있다. 즉, 2개의 클라이언트를 생성하여 학습을 진행한 A3C 알고리즘은 2,000분 이내에서 학습이 수렴되지는 않았지만, 평균 보상과 평균 품질이 점차 증가하는 것을 볼 수 있다. 그리고 4개의 클라이언트를 사용한 경우는 약 1,020분 정도 비디오를 재생했을 때, 8과 9의 품질을 갖는 세그먼트가 선

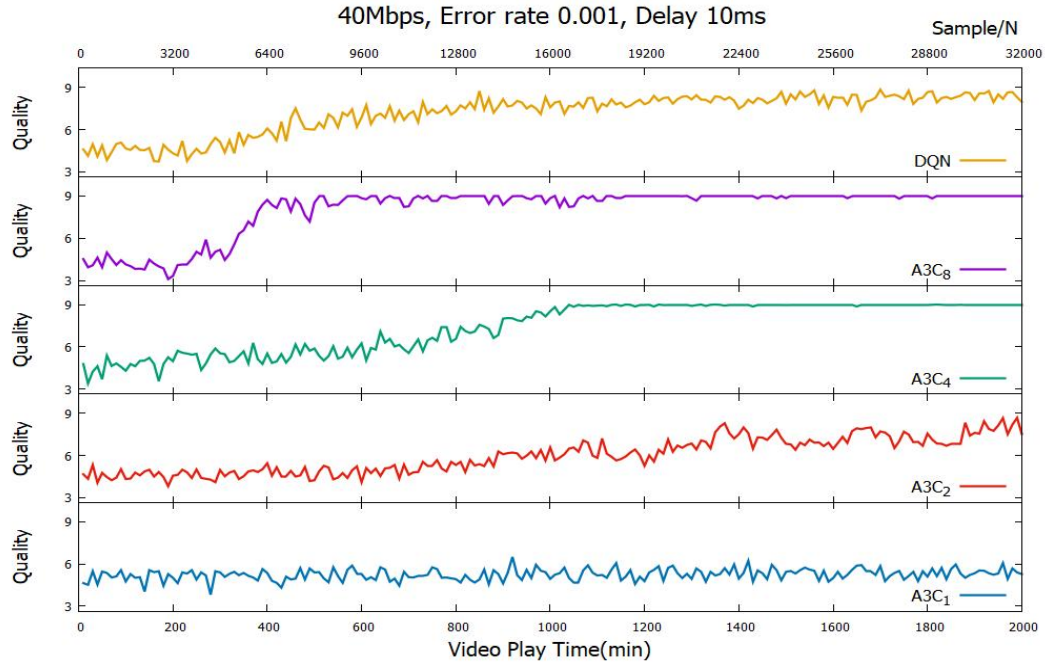

그림 5. 비디오 재생 시간에 따른 평균 세그먼트 품질의 변화, (40, 0.001, 10)
Fig. 5. Changes of Average Segment Quality over Video Play Time, (40, 0.001, 10)

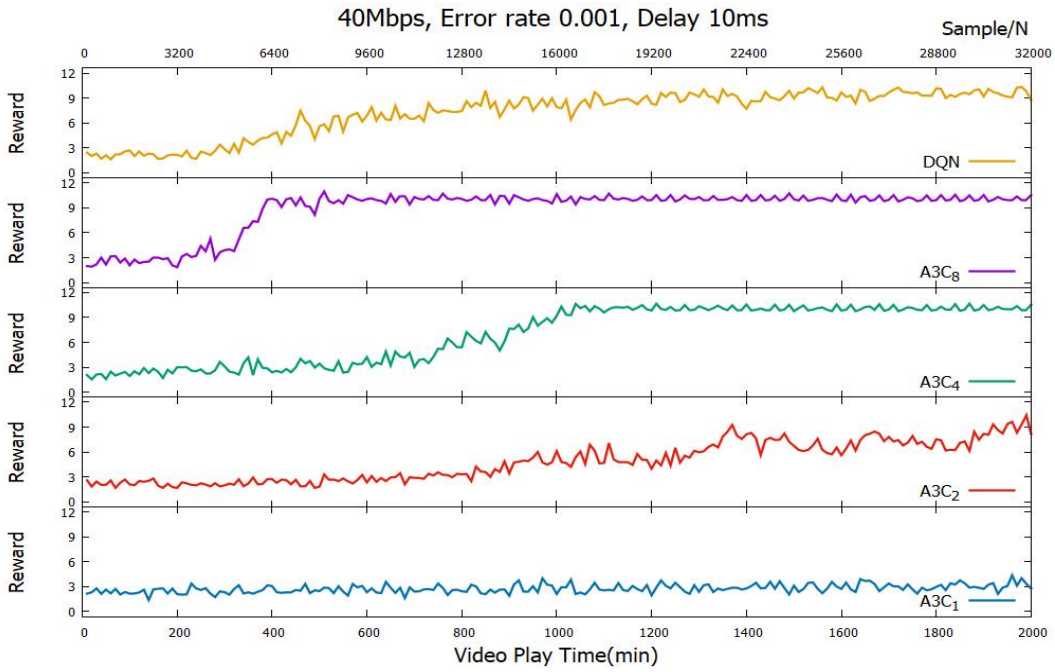

그림 6. 비디오 재생 시간에 따른 평균 보상의 변화, (40, 0.001, 10)
Fig. 6. Changes of Average Reward over Video Play Time, (40, 0.001, 10)

택되도록 학습이 수렴되었다. 더 나아가 8개의 클라이언트를 사용한 경우는 학습곡선의 기울기가 가파르게 증가 되었으며 약 400분에서 500분 정도의 비디오를 재생했을 때 8과 9의 품질을 갖는 세그먼트가 선택되도록 학습이 수렴되었다. 이러한 결과들로 A3C 알고리즘은 클라이언트 수의 증가에 비례하여 학습속도가 빠르게 증가하는 것을 확인할 수 있었으며 4개 이상의 클라이언트를 사용한 A3C 알고리즘은 DQN 알고리즘과 비교하여 월등히 좋은 성능을 나타냄을 알 수 있었다.

두 번째 환경은 손실률이 0.01이고 지연시간이 100ms로

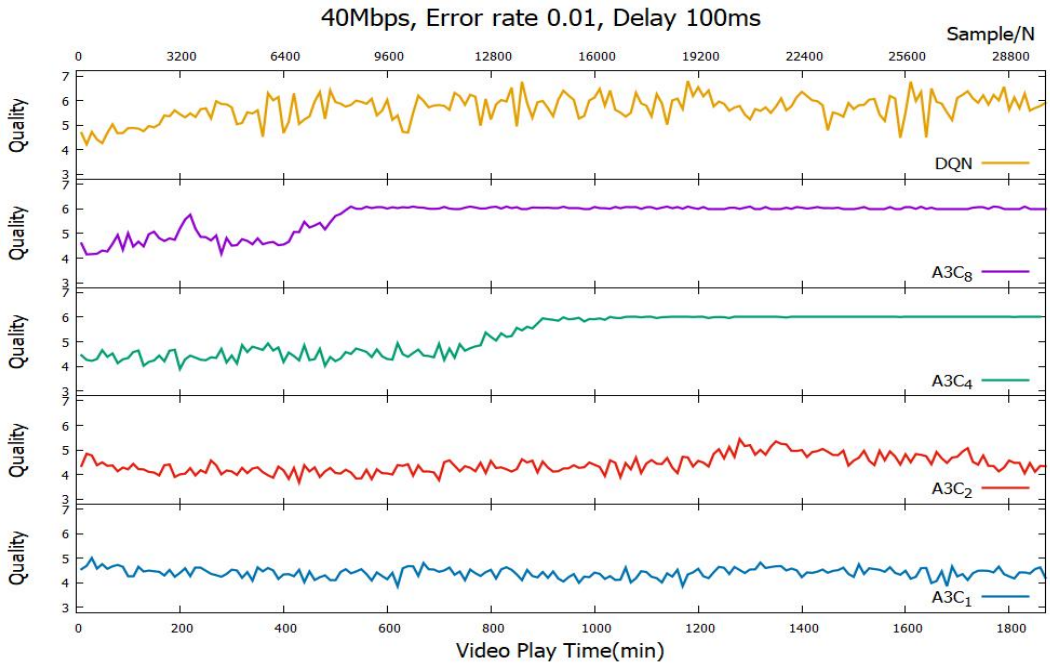

그림 7. 비디오 재생 시간에 따른 평균 세그먼트 품질의 변화, (40, 0.01, 100)
Fig. 7. Changes of Average Segment Quality over Video Play Time, (40, 0.01, 100)

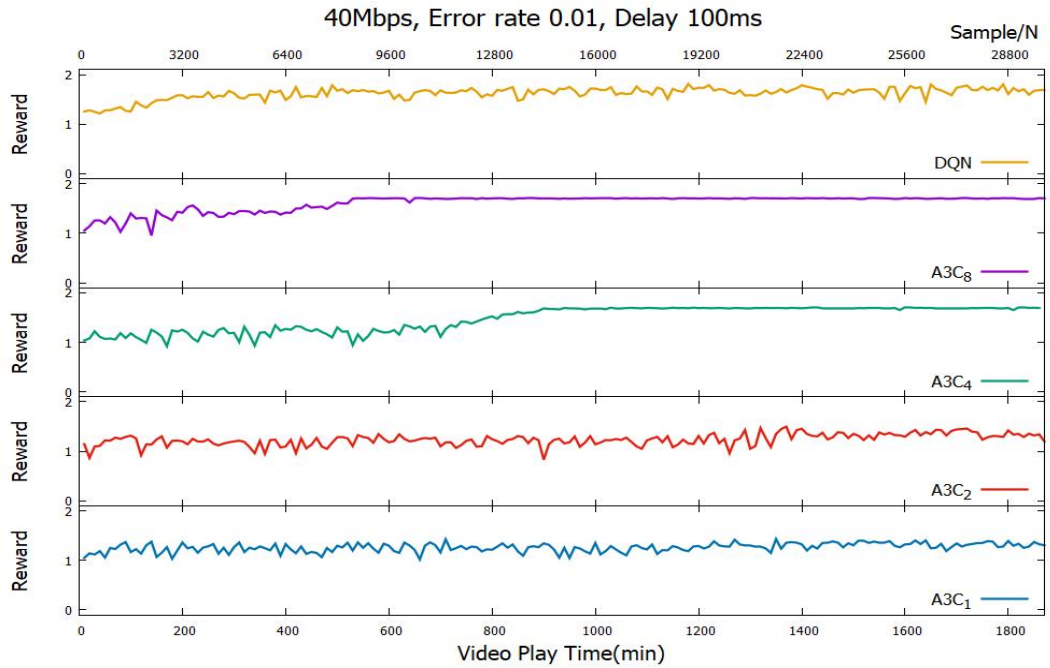

그림 8. 비디오 재생 시간에 따른 평균 보상의 변화, (40, 0.01, 100)
Fig. 8. Changes of Average Reward over Video Play Time, (40, 0.01, 100)

손실률과 지연시간이 비교적 높은 환경이다. 이 환경에서 클라이언트는 최고 품질의 비디오를 재생할 수는 없지만 보통 5와 7 사이의 품질을 갖는 세그먼트를 원만하게 재생할 수 있으므로 이에 적절하도록 학습모델이 생성되어야 한다. 그림 7과 그림 8을 보면 재생시간이 충분하지 않은 학습 초기에는 모든 알고리즘이 네트워크 상태에 적합한 품질을 선택하지 못하다가 시간이 지날수록 상응하는 품질을 선택하도록 학습이 되고 평균 보상 또한 증가하는 것을 볼 수 있

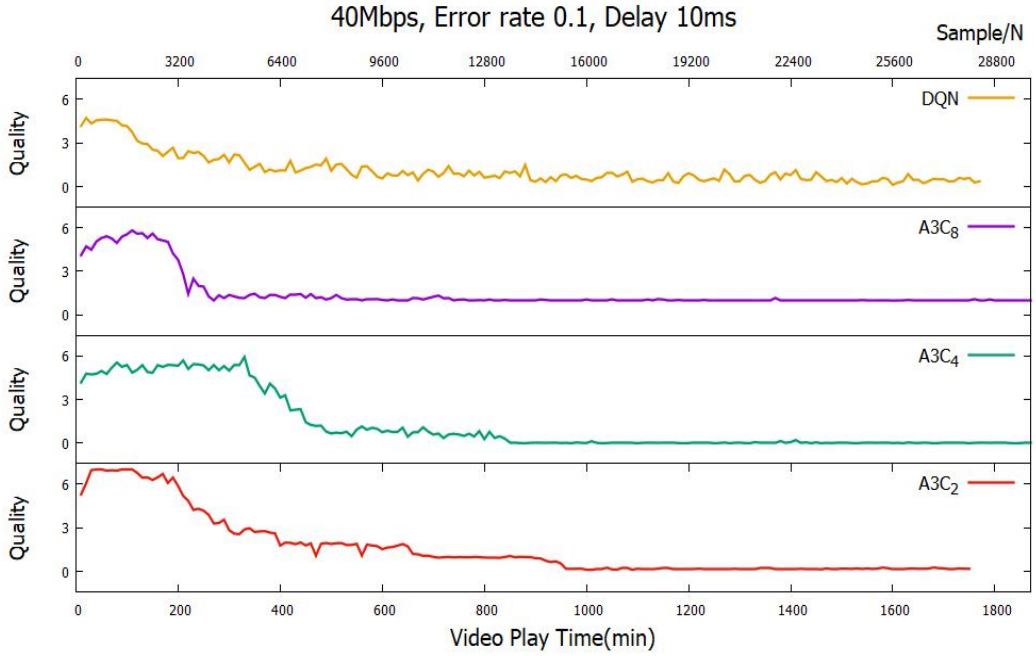

그림 9. 비디오 재생 시간에 따른 평균 세그먼트 품질의 변화, (40, 0.1, 10)
Fig. 9. Changes of Average Segment Quality over Video Play Time, (40, 0.1, 10)

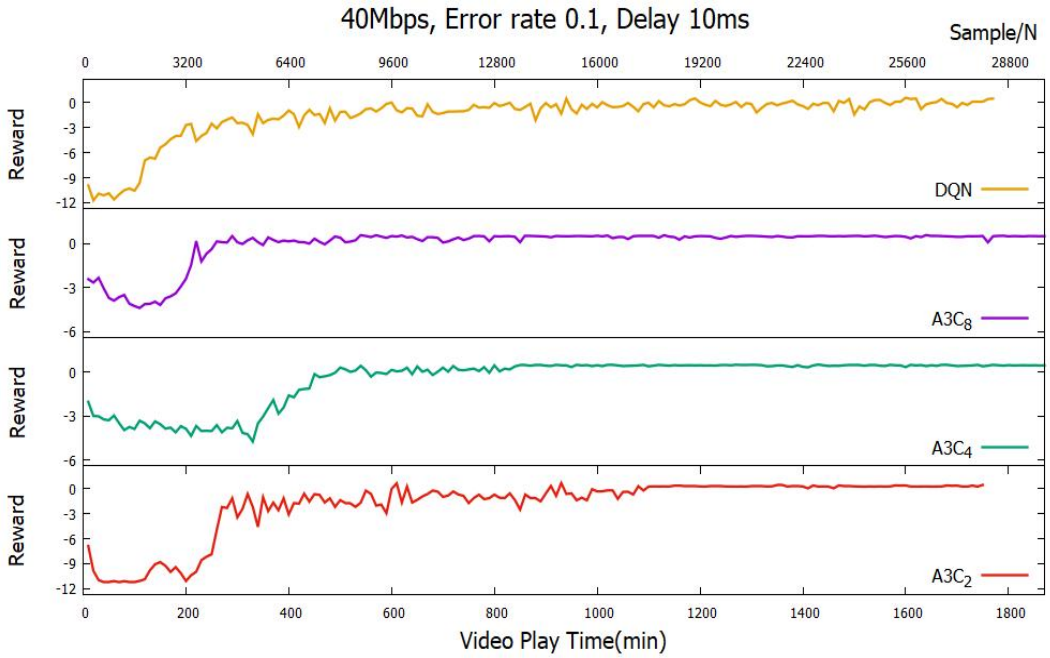

그림 10. 비디오 재생 시간에 따른 평균 보상의 변화, (40, 0.1, 10)
Fig. 10. Changes of Average Reward over Video Play Time, (40, 0.1, 10)

다. 이 환경은 비교적 긴 지연시간에 에러율이 높은 네트워크 환경인데 이 경우에 DQN 알고리즘은 평균 보상이 안정적인 선을 나타내는 구간이 있으나 선택하는 품질이 4와 7 사이로 변화폭이 큰 것을 볼 수 있다. 학습이 수렴되는 재생 시간은 500분 정도이며 이는 단일 클라이언트로 50회 정도 비디오를 재생해야 하는 시간이다. 1개의 클라이언트를 생성하여 학습을 진행한 A3C 알고리즘 역시 학습이 잘 수렴되지 않는 모습을 보인다. 2개의 클라이언트를 생성하여 학습을 진행한 A3C 알고리즘 또한 학습곡선의 기울기가 커지는 것은 확인할 수 있었으나, 비디오 재생시간 2,000분이

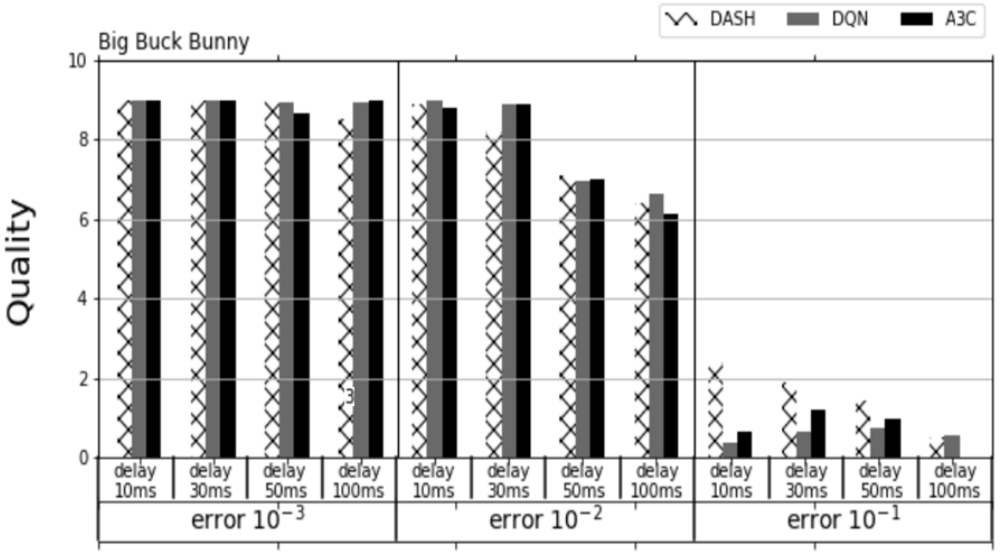

그림 11. 네트워크 환경의 변화에 따른 평균 세그먼트 품질
Fig. 11. Average Segment Quality According to Network Conditions

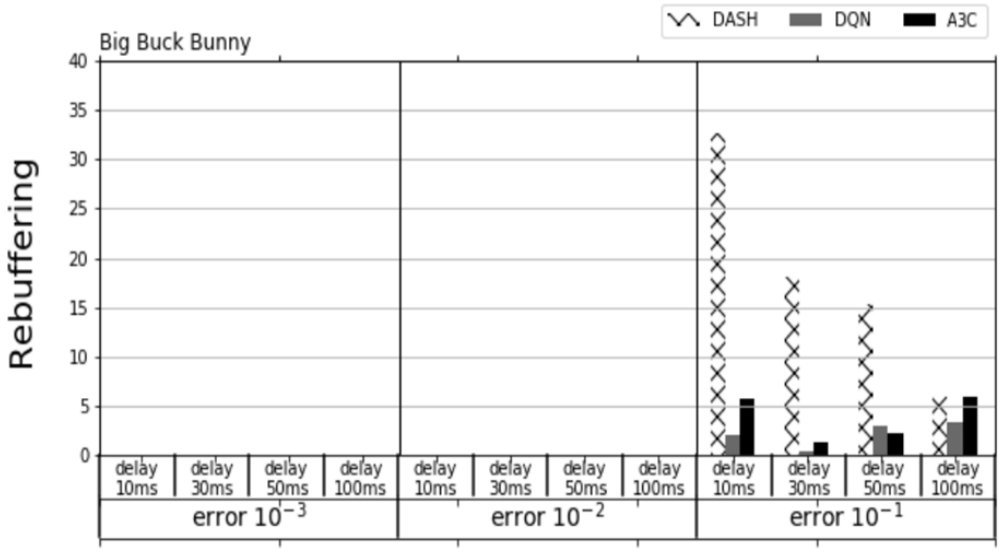

그림 12. 네트워크 환경의 변화에 따른 리버퍼링 횟수
Fig. 12. The Number of Rebuffering According to Network Conditions

지나도 학습이 수렴되지 않았다. 4개의 클라이언트를 생성한 A3C 알고리즘부터 5와 6 사이의 품질을 선택할 수 있도록 학습이 수렴되었으며 그에 따라 평균 보상도 증가하였다. 4개의 클라이언트를 사용했을 때, 비디오를 900분 정도 재생하면 학습이 수렴되었고 8개의 클라이언트를 사용했을 때는 550분 정도 재생하면 학습이 수렴되었다. 손실률과 지연시간이 비교적 높은 이러한 환경에서도 A3C 알고리즘은 클라이언트 수의 증가에 비례하여 학습속도가 증가하는 것을 확인할 수 있었다.

세 번째 환경은 손실률이 0.1로 매우 높아 0과 1사이의 저품질 세그먼트가 아니면 리버퍼링 없이 스트리밍하기 어려운 환경이다. 따라서 비디오 클라이언트는 저품질을 서버로 요청해야 한다. 그림 9와 그림 10을 보면 재생시간이 충분하지 않은 학습 초기에는 네트워크 환경에 적합한 비디오 품질을 선택하지 못하여 리버퍼링이 발생하고 평균 보상이 음수값이 나온다. 보상 계산식을 참고하면 리버퍼링이 발생했을 경우만 음수 (-)의 보상을 주었다. DQN과 A3C는 비디오 재생시간이 증가하며 점점 0~1품질을 선택하도록 학습

이 진행된다. DQN의 학습곡선은 비디오 재생시간 약 900분 정도에서 학습이 수렴되는 것을 확인할 수 있다. A3C 기반의 메커니즘은 이 환경에서도 클라이언트 수에 비례하여 학습속도가 증가하는 것을 확인할 수 있었다. 2개의 클라이언트를 사용했을 때 970분, 4개의 클라이언트를 사용했을 때 830분 그리고 8개의 클라이언트를 사용했을 때 300분 정도에서 학습이 수렴되는 것을 확인할 수 있었다.

2.2 네트워크 환경 변화에 따른 시스템 성능

기존의 DASH ABR 알고리즘과 DQN과 A3C 기반 알고리즘의 전체적인 성능을 비교하기 위하여 DQN과 A3C는 모든 환경에서 안정적으로 수렴된 학습모델을 사용하여 실험을 진행하였다. 그림 11과 그림 12는 세 종류의 에러율(0.1, 0.01, 0.001)과 네 종류의 지연시간(10ms, 30ms, 50ms, 100ms)을 조합하여 설정한 12가지 네트워크 환경하에서 위의 세 가지 알고리즘을 사용하는 각 경우에 대하여 평균 세그먼트 품질과 리버퍼링 횟수를 나타낸다.

에러율이 0.001로서 비교적 좋은 네트워크 환경에서는 위의 세 가지 알고리즘 모두 8과 9사이의 고품질 세그먼트를 선택하면서 리버퍼링이 발생하지 않는 양질의 스트리밍 서비스가 가능함을 확인할 수 있었다. 그러나 에러율이 0.01인 네트워크 손실이 비교적 많이 발생하는 경우에는 지연시간이 10ms 혹은 30ms인 환경, 즉 근거리 환경에서는 여전히 높은 품질의 스트리밍 서비스가 가능하지만, 지연시간이 50ms 이상 100ms가 되는 원거리 환경에서는 위의 세 가지 알고리즘 모두 비디오 품질이 6과 7 사이로 떨어짐을 확인할 수 있었다. 그러나 리버퍼링이 발생하지 않는 것은 나빠진 네트워크 환경에서도 여전히 잘 적응하고 있다는 것을 의미한다. 무선 네트워크 환경과 같이 에러율이 0.1로서 아주 높은 경우에는 기존 DASH ABR 알고리즘은 DQN과 A3C 기반 알고리즘보다 비교적 높은 비디오 품질을 선택했고 이에 따라 리버퍼링 횟수가 급격히 높아지는 것을 확인할 수 있었다. 반면에 DQN과 A3C 기반 알고리즘은 네트워크 환경에 맞도록 비교적 낮은 비디오 품질을 선택하여 리버퍼링 횟수를 크게 줄이는 것을 확인할 수 있었다.

V. 결 론

본 논문에서는 적응형 비디오 스트리밍을 위한 분산 강화학습 기반의 비디오 품질 선택 메커니즘을 제안하였다. 유무선 동적 네트워크 환경에서 끊김 없는 비디오 스트리밍을 위해 DASH가 제안되었고, 그 후 DASH의 성능을 개선한 많은 연구가 진행되어왔다. 그러한 노력의 일환으로 DQN 기반의 적응형 비디오 스트리밍이 제안되었으나 이 기법은 많은 학습데이터와 시간이 필요하다는 단점이 있었다. DQN은 학습데이터 간의 높은 상관관계를 해결하기 위하여 리플레이 메모리를 사용하였고, 이는 메모리 공간을 낭비한다. 또한, 기존 DQN 기반의 적응형 비디오 스트리밍 시스템은 클라이언트와 일대일로 학습하기 때문에 서버-클라이언트 구조인 비디오 스트리밍 시스템에서 클라이언트 수의 증가에 따른 다양한 학습데이터를 활용할 수 없었다. 본 논문에서는 이러한 단점을 해결하기 위해서 A3C 기반의 적응형 비디오 스트리밍 시스템을 제안하였다. 이 시스템은 비디오 서버와 클라이언트, A3C 서버로 구성되어 있는데 비디오 서버와 클라이언트는 network simulator-3 (ns-3)로 구현하였고 A3C 서버는 케라스로 구현하였다. 구현된 시스템의 성능을 평가하기 위하여 유무선 복합 네트워크의 다양한 환경을 반영하기 위하여 모두 12가지 네트워크 환경을 설정하고 실험을 진행하였다. 실험 결과로 A3C 기반 알고리즘이 기존의 DASH ABR 알고리즘보다 다양한 네트워크 환경에 보다 적합한 비디오 품질을 선택할 확률이 높음을 확인할 수 있었다. 또한, 독립적이고 상이한 네트워크 환경을 가지는 다수의 클라이언트로부터 다양한 학습데이터를 수집하고 비동기적으로 하나의 학습모델을 생성함으로써 클라이언트 수가 증가함에 따라 네트워크 환경에 빠르게 적응하는 것을 확인할 수 있었다. 향후 연구에서는 A3C 기반의 적응형 스트리밍 시스템에서 다양한 세그먼트 크기를 가지는 비디오를 학습시켜 비디오 컨텐츠의 내용이 성능에 미치는 영향을 연구할 예정이다. 또한, 클라이언트 수를 8개 이상으로 증가시켜 학습속도의 변화를 측정하고 최적의 학습속도를 낼 수 있는 클라이언트 수를 확인하고자 한다.

References

[1] J. Kua, G. Armitage, P. Branch, “A Survey of Rate Adaptation Techniques for Dynamic Adaptive Streaming Over HTTP,” IEEE Communications Surveys and Tutorials, Vol. 19, No. 3, pp. 1842-1866, 2017.

[2] K. Miller, E. Quacchio, G. Gennari, A. Wolisz, “Adaptation Algorithm for Adaptive Streaming over HTTP,” 2012 IEEE 19th International Packet Video Workshop, pp. 173 - 178, 2012.

[3] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoßfeld, P. Tran-Gia, “A Survey on Quality of Experience of HTTP Adaptive Streaming,” IEEE Communications Surveys & Tutorials, Vol. 17, No. 1, pp. 469-492, 2015.

[4] I. S. Kim, S. Hong, S. Jung, K. Lim, “An Intelligent Video Streaming Mechanism based on a Deep Q-Network for QoE Enhancement,” Journal of Korea Multimedia Society, Vol. 21, No. 2, pp. 188-198, 2018.

[5] I. S. Kim, K. Lim, “The Effect of Segment Size on Quality Selection in DQN-based Video Streaming Services,” Journal of Korea Multimedia Society, Vol. 21, No. 10, pp. 1182-1194, 2018.

[6] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Harley, T. P. Lillicrap, D. Silver, K. Kavukcuoglu, “Asynchronous Methods for Deep Reinforcement Learning,” Proceedings of The 33rd

International Conference on Machine Learning, PMLR 48:1928-1937, 2016.
[7] T. Haarnoja, A. Zhou, P. Abbeel, S. Levine, "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor," Proceedings of the 35th International Conference on Machine Learning, PMLR 80:1861-1870, 2018.
[8] P. Juluri, V. Tamarapalli, D. Medhi, "QoE Management in DASH Systems Using the Segment Aware Rate Adaptation Algorithm," Proceeding of IEEE/IFIP Network Operations and Management Symposium, pp. 129-136, 2016.
[9] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, "Playing Atari with Deep Reinforcement Learning," NIPS Deep Learning Workshop 2013, arXiv preprint arXiv:1312.5602, 2013.
[10] T. R. Henderson, M. Lacage, G. F. Riley, "Network Simulations with the ns-3 Simulator," Proceeding of Association for Comput ing Machinery Conference on Special Interest Group on Data Communication, pp. 527, 2008.

Minje Choi (최 민 제)

2019 Computer Information Engineering from Daegu University (B.S.)
2021 Computer Science and Engineering from Kyungpook Natl. Univ. (M.S.)
Fields of Interest: Mobile Computing, Artificial Intelligence
Email: alswp25@gmail.com

Kyungshik Lim (임 경 식)

1982 Electronics from Kyungpook Natl. Univ. (B.S.)
1985 Computer Science from KAIST (M.S.)
1994 Computer Science from Univ. of Florida at Gainesville (Ph.D.)
Fields of Interest: Mobile Computing, Artificial Intelligence
Email: limkyungshik@gmail.com