

실시간 스트리밍을 위한 서버 협업 태스크 오프로딩 시스템 설계

이종혁, 서효운
광운대학교 전자통신공학과

Ljh000331@kw.ac.kr, hyowoonseo@kw.ac.kr

Designing Server-Cooperative Task Offloading System for Realtime Streaming

Jong Hyuk Lee, HyoWoon Seo
Dept. Electronics and Communications Eng., Kwangwoon Univ.

요 약

본 논문은 실시간 스트리밍 중 제한된 컴퓨팅 자원으로 인한 QoS 품질 하락 방지 목적 서버 협업 태스크 오프로딩 시스템 성능에 대한 실험 결과를 보여준다. 실시간 스트리밍 환경에서 가장 중요한 컴퓨팅 자원은 CPU 성능이며 CPU 는 사용량이 70%가 넘어갈 때 과부하 상태로 판단되어 연산 속도가 느려진다. 따라서 CPU 사용량에 따라 실시간 스트리밍 QoS 가 결정되므로 본 논문에서는 실시간 스트리밍을 하는 모든 엣지 서버의 CPU 사용량이 60%가 넘어가지 않도록 엣지 서버끼리 처리할 데이터를 나눠 가지도록 설계한 시스템을 작동 순서에 맞추어 보여준다.

I. 서 론

ICT 기술이 발전함에 따라 발생하는 방대한 데이터 양으로 인한 데이터 트래픽이 다양한 분야에서 문제로 대두되었다. 그 중에서도 특히 실생활에서 느낄 수 있을 정도로 데이터 트래픽으로 인해 심각한 피해를 받는 분야는 OTT 플랫폼이다. 이 분야의 대표적인 기업들조차 스트리밍 전송률, 화질에 제한을 두고 있으며 높은 수준의 QoS 품질을 만족하는 기술이 개발되었음에도 데이터 트래픽으로 인해 오히려 퇴화 중인 상황에 이르렀다. 이러한 문제를 해결하고자 다양한 태스크 오프로드 기법이 탄생하고 있다. 본 논문은 태스크 오프로드 연구에 기여하고자 실시간 스트리밍 환경을 위한 서버 협업 태스크 오프로딩 시스템을 제안하고 실험 결과를 보인다.

II. 본론

태스크 오프로딩이란 클라우드 컴퓨팅, 엣지 컴퓨팅을 포괄한 상위 개념으로 제한된 컴퓨팅 자원으로부터 컴퓨팅 자원이 풍부한 컴퓨팅 플랫폼으로 태스크를 전달함으로써, 한계 보다 더 많은 데이터를 효율적으로 연산 및 처리하도록 하는 개념이다. 즉, 컴퓨팅 자원 상태를 나타낼 수 있는 엣지 서버와 컴퓨팅 자원 기반으로 태스크를 분배하는 제어 역할 중앙 서버를 구성하였다.

	개 수	역할	용도
Laptop	3	중앙 서버-1 엣지 서버 -2	엣지 컴퓨팅 환경 구성
IP 카메라	3	데이터 발생 장치	로드밸런서 구현
Wifi 공유기	1	무선연결환경	로드밸런서 구현

Figure 1 - 태스크 오프로딩 시스템 구성장비

그림 1 은 실시간 스트리밍 환경에서 태스크 오프로딩 시스템을 설계하기 위한 구성장비이다. 중앙서버는 IP 를 기반으로 엣지 서버와 처리해야할 작업을 구별하기 때문에 IP 를 기준으로 트래픽을 분산시키는 L4 로드밸런서 역할을 한다. 그리고 실시간 스트리밍 환경에서의 태스크 오프로딩을 구현하기 위해 Rtsp 프로토콜을 준수하여 Wifi 공유기를 매개체로 모든 서버와 IP 카메라가 같은 네트워크 망을 이루도록 하였다.

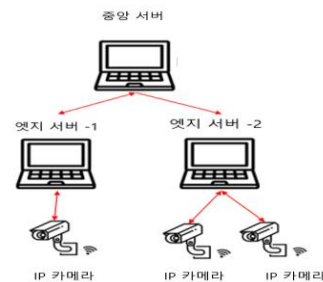


Figure 2 - 서버 협업 태스크 오프로딩 시스템 모식도

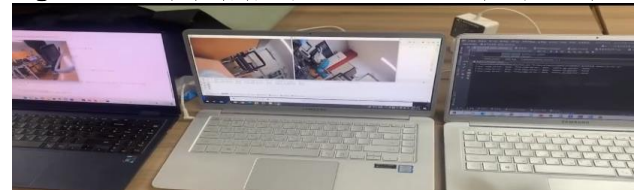


Figure 3- 서버 협업 태스크 오프로딩 시스템 실험

그림 2 와 그림 3 은 시스템의 초기상태를 나타낸다. 엣지 서버 1 에 IP 카메라가 한 대 무선 연결되어 있으며 엣지 서버 2 엔 IP 카메라가 두 대 무선연결 되어 있다. 마지막으로 중앙 서버와 2 개의 엣지 서버는 무선으로 연결되어 있음을 보여준다. 이 때, 각각의 IP 카메라가

나타내는 업무량을 다른 수치로 설정하여 시스템을 구성하였다.

이와 같은 구조를 가진 서버 협업 태스크 오프로딩 시스템에서 컴퓨팅 자원을 기반으로 최적의 부하 분산을 구현하기 위해선 각 엣지 서버의 컴퓨팅자원 상태와 어떤 IP 카메라가 무선연결 되어 있는지에 대한 정보를 실시간으로 추출해야 한다. 뿐만 아니라 엣지 서버는 컴퓨팅 자원 정보를 실시간으로 추출하는 동시에 로드 밸런서 역할을 하는 중앙서버에 추출된 정보를 전송해야 한다.

중앙 서버가 로드 밸런서 역할을 하는 이유는 엣지 서버들의 컴퓨팅 자원을 기반으로 가장 최적의 부하 분산 상태를 만들도록 하는 협업 명령이 IP 정보로 이루어져 있기 때문이다. 그리하여 엣지 서버가 중앙 서버에게 컴퓨팅 자원 정보를 실시간 전송하고 중앙 서버는 엣지 서버에게 협업 명령 정보를 실시간 전송하여 실시간 양방향 정보 교환이 이루어지므로 소켓 프로토콜을 준수하였다.

여러 개의 엣지 서버 컴퓨팅 자원 정보를 실시간으로 받는 중앙 서버는 CPU 사용량에 대한 정보를 이용하여 로드 밸런싱을 하도록 설계하였다. 보통 CPU 사용량이 70%가 넘어가면 CPU 과부하 상태로 판단되고 연산 속도, 데이터 처리 속도가 줄어든다. 이러한 이유로 CPU 사용량이 클수록 실시간 스트리밍 QoS 품질이 하락하게 된다. 그리하여 여러 컴퓨팅 자원 중 CPU 사용량을 기준으로 엣지 서버들에게 부하를 분산하도록 설계하였다. 이 때, 엣지 서버들의 CPU 성능은 동일하고 IP 카메라로부터 발생하는 작업량으로 인한 CPU 사용량은 각각의 IP 카메라마다 다르도록 설정했기 때문에 이를 고려한 라운드 로빈 방식을 기반으로 부하 분산 알고리즘을 설계하였다. 추가로 CPU 사용량이 60% 이상인 엣지 서버가 탐지되면 알고리즘이 작동하도록 설계했기 때문에 과부하를 미리 방지하여 최대한 엣지 서버가 과부하 상태가 되지 않도록 하였다.

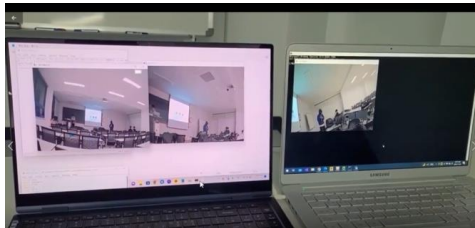


Figure 4 - 엣지 서버 2 과부하 발생

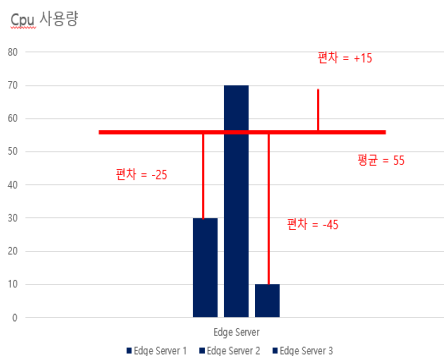


Figure 5 - 부하분산 알고리즘 계산과정

중앙 서버가 엣지 서버의 컴퓨팅 자원을 실시간으로 모니터링 하는 도중 그림 4 와 같이 임의의 엣지 서버가 과부하 직전 상태가 된다면 그림 5 와 같이 중앙 서버에서 라운드 로빈 기반 부하 분산 알고리즘 계산을 시작한다. 모든 엣지 서버의 CPU 사용량을 기반으로 평균을 내어 각각의 엣지 서버의 편차를 계산한다.

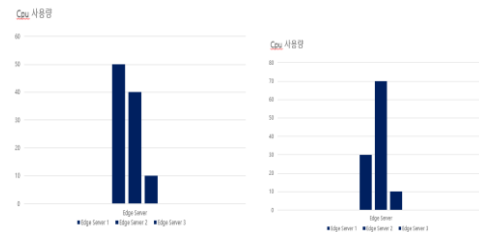


Figure 6 - 최적의 부하 분산 방법 탐색

그림 6 은 중앙 서버가 서로 다른 작업량을 요구하는 IP 카메라가 어떤 엣지 서버에 연결을 맺어야 부하 분산이 가장 균등하게 이루어지는지 연산하는 과정을 그림으로 나타낸 것이다. 중앙 서버는 연산이 끝나면 엣지 서버간 어떻게 협력해야 하는지에 대한 정보를 엣지 서버들에게 브로드캐스트를 하여 엣지 서버들이 이행하고 그림 7 과 같은 상황을 만들어 낸다.

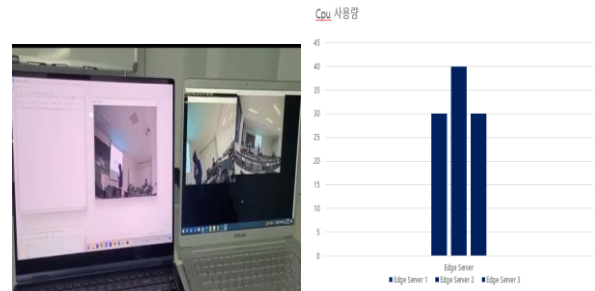


Figure 7 - 중앙 서버가 서버 협력을 명령한 이후

III. 결론

본 논문에서는 엣지 서버간 최적의 부하 분산을 항상 유지하는 서버 협력 기법을 어떻게 설계하였는지 설명하였다. 시연을 하며 컴퓨팅 자원과 서버 협력 명령을 Json 파일, 자바 스크립트로 실시간 모니터링 하며 설계 목표대로 잘 구현됨을 확인할 수 있었다. 하지만 서버 협력이 시작될 때 IP 카메라의 무선 연결이 끊어진 뒤 다른 엣지 서버로 무선 연결이 맺어질 때 한 순간 CPU 사용량이 70%가 넘게 된다. 만약 그 순간이 컴퓨팅 정보를 중앙 서버에 보내는 스케줄과 타이밍과 일치한다면 그에 따른 부하 분산 알고리즘이 작동하는 경우가 드물게 있었다. 이는 라즈베리 파이와 같은 고성능 HW 를 사용하면 발생하지 않는 오류라는 결론을 지을 수 있다.

ACKNOWLEDGMENT

이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.NRF-2022R1F1A1075078).

참 고 문 헌

- [1] Lee, Eun-hee, Lee, Su-kyung "Task Offloading Algorithm for Mobile Edge Computing", 한국통신학회논문지, 2021.02
- [2] S.G.Choi, B.H.Ryu,C.Y.Lee,E.B.Kim,J.H.Na "Data Offloading Techniques in 3GPP" 한국전자통신연구원,2015.08.01