

모바일 환경에서 VOD 서비스를 위한 프락시간의 효율적인 로드밸런싱 기법

김현경, 백창현, 정홍기, 박승규

아주대학교 정보통신 전문 대학원

전화 : 031-219-2532 / 핸드폰 : 011-9353-8269

An effective load balancing among proxies for VOD service in mobile environments

Hyun-Kyung Kim, Chang-Hyun Baek, Hong-Ki Jung, Seung-Kyu Park

Graduate School Information and Communication, Ajou University

E-mail : pinkp@madang.ajou.ac.kr

Abstract

The transcoding is a method which converts a multimedia stream to an adaptive format, such as frame rate and resolution, that can be played in the mobile devices. Due to the heavy consumption of CPU power for transcoding, the server in charge cannot supply all streams requested from clients if it is already occupied by several streams. The load balancing among proxies is a common approach for solving such problem. In addition to collaborative works for load balancing, the mobility must be taken into account in the mobile environments. Thus, the wired-network based algorithm is not suitable for the streams service in the mobile environments. In this paper, we propose an alternative scheme for cooperative proxy architecture that considers the case in which mobile devices are moving over other zones.

I. 서론

무선 네트워크 기술의 발달로 인해 사용자는 기존 유선 환경에서 이용하던 VOD(Video On Demand) 서비스를 노트북, PDA, 핸드폰 등과 같은 단말기를 통하여 무선 환경에서도 제공받을 수 있게 되었다. 그러나 무선 환경에서는 제한된 대역폭과 CPU 처리능력 같은

일반적인 취약성과 다양한 단말기 각각의 특성들에 의해, 서버에서 제공하는 멀티미디어 스트림을 무선 환경에서 원활하게 제공하기 어렵다. 이런 문제점을 해결하기 위한 방법으로 각각의 무선 단말기에서 재생이 가능한 적절한 형태(frame rate, resolution 등)의 멀티미디어 스트림으로 변환하는 트랜스코딩 작업이 필요하다. 그러나 트랜스코딩 작업을 수행하기 위해서는 많은 CPU 자원의 소모가 요구되기 때문에 하나의 서버에서 동시에 처리가능 한 트랜스코딩 작업의 개수는 제한적이다. 제한된 자원을 이용하여 사용자의 요청에 적합한 서비스를 제공하기 위해서서서 서버의 CPU 작업 부하를 분산시키는 방법이 요구되는데, 가장 일반적인 방법으로는 프락시 서버를 이용하는 기법이 있다. 프락시 서버를 사용하면 서버의 작업 부하를 프락시로 분산시켜 더 많은 수의 트랜스코딩 작업을 수행 할 수 있으며, 캐쉬에 트랜스코딩 된 비디오 데이터를 저장하여 사용자의 요청이 있을 경우 저장된 데이터를 서비스 하여 트랜스코딩 작업의 수를 줄일 수 있다. 하지만 프락시 기반의 로드밸런싱과 같은 기존에 유선 환경의 특성을 고려하여 만들어진 로드밸런싱 기법은 사용자의 이동성이 강조되는 무선 환경에 적용시키기에는 적합하지 않다. 따라서 본 논문에서는 상호 협동하는 프락시 구조 하에서 CPU자원의 사용률과 사용자의 이동성을 함께 고려한 효율적인 모바일 로드밸런싱 (MLB : Mobile Load Balancing) 기법을 제안하고, 시뮬레이션을 통하여 기존의 방식과 제안된 방식을 비교하여 성능이 향상됨을 보인다.

II. 관련연구

2.1 트랜스코딩

무선 환경에서 원활한 VOD서비스를 위해서는 스트림으로 전송되는 비디오 데이터가 단말기의 특성에 적합한 이미지 사이즈와 프레임 비율 등을 가져야한다. 이런 요청에 해당하는 변환을 트랜스 코딩 작업이라고 한다.

일반적인 트랜스코딩 기법은 트랜스코딩 작업을 수행하는 위치에 따라 서버 기반의 트랜스코딩 기법과, 프락시 기반의 트랜스코딩 작업이 있다. 특정 서버나 프락시가 트랜스코딩을 담당할 경우 네트워크의 부하나 CPU의 부하가 고려되지 않는다는 단점이 있다. 이런 문제를 해결하기 위해 Xueyan Tang et. al.은 네트워크 대역폭과 CPU의 power를 고려하여 트랜스코딩 작업을 서버에서 수행할지 프락시에서 수행할지 결정하여 서버와 프락시간의 로드밸런싱과 서버 쪽의 네트워크의 부하를 감소시키는 방법을 이용하였다.[1]

본 논문에서는 프락시의 사용을 확대하고, CPU의 사용률과 사용자의 이동성을 함께 고려한 트랜스코딩을 작업을 하는 방법을 제안한다.

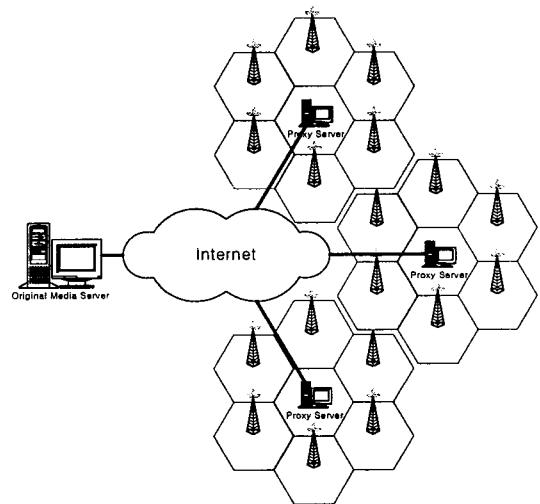
2.2 프락시 캐쉬

프락시 캐쉬은 서버에서 제공하는 서비스의 일부를 사용자와 가까운 곳의 프락시의 캐쉬에 저장하여, 사용자의 요청이 있을 때 서버에 서비스를 요청하여 서비스를 제공받는 대신 캐쉬에 저장된 데이터를 사용자에게 서비스 하여주는 방법으로서 서버 로드와 서비스 대기 시간을 감소시킬 수 있다. 하지만 VOD 서비스 같은 경우 일반 웹 페이지보다 수천만 배 더 크기 때문에 캐쉬에 저장 할 수 있는 양이 극히 제한적이다. 이런 문제 해결을 위해 많은 기법들이 연구되었는데 대표적인 기법으로는 멀티미디어 데이터의 앞부분만을 캐싱하여 시작 지연 시간을 줄이 수 있는 Prefix 캐싱 기법과, 같은 영화를 보는 사람들을 위해 데이터를 캐쉬 내에서 삭제하지 못하게 하는 Interval 캐싱 기법, 프락시 캐싱 서버를 분산시켜 프락시 서버간의 데이터 교환을 통해 캐싱 공간 최대화하고 응답시간을 최소화하는 프락시 간의 상호협동기법들이 있다.

III. System Architecture

본 논문에서 제안된 Mobile load balancing 작업을 위한 전체 시스템 구성은 [그림 1]과 같다. 이 구조에

서 서버와 프락시는 유선 네트워크로 연결되어 있으며, 프락시와 무선 단말기 사용자 사이에는 베이스 스테이션(BS)이 위치하여 프락시에서 트랜스코딩 된 데이터를 유선 네트워크를 통해 전송받고, 전송 받은 데이터를 무선 네트워크에 연결되어 있는 사용자에게 서비스 한다.



[그림 1] System Architecture

서버에는 모든 비디오의 원본 스트림을 가지고 있으며, 프락시의 요청에 의해 비디오 스트림을 제공한다.

프락시는 제공된 비디오 스트림을 사용자의 무선 단말기에 알맞은 형태로 변환하는 작업을 수행하며, 동시에 다른 사용자에 의해 미래에 사용될 것을 대비하여 프락시 내의 캐쉬에 트랜스코딩 된 스트림을 저장한다.

무선 단말기 사용자는 프락시에게 비디오 스트림을 요청할 때 자신의 단말기 정보를 함께 보내어 서비스하기에 적합한 데이터를 요구하게 된다.

무선 단말기 사용자에 의한 비디오 스트림의 새로운 요청은, 베이스 스테이션을 통해 자신이 연결되어 있는 프락시에게 전해지며, 요청을 받은 프락시에서는 사용자의 무선 단말기에 대한 정보를 얻어 와서, 단말기에 적합한 형태의 트랜스코딩 된 스트림이 자신의 캐시에 존재하는지를 검사하여 자신의 캐쉬 내에 존재한다면, 그 즉시 사용자에게 제공한다. 그러나 존재하지 않는 경우에는 인접한 프락시를 검사하여, 인접한 프락시 내에 요청한 스트림이 존재하면, 그곳으로부터 스트림을 다운받아서 사용자에게 제공하고, 인접한 프락시 내에도 요청한 스트림이 존재하지 않는다면, 서버로부터 원본 스트림을 다운받아 트랜스코딩 작업을 수행 한 후 사용자에게 제공한다. 만약 트랜스코딩을 할 수 있는 CPU 자원이 남아 있지 않은 경우에는 인접한 프락시

로 작업을 분산시켜야 하며, 로드 분산을 위해 4장에 제시된 Mobile Load Balancing 알고리즘을 통하여 트랜스코딩 작업을 수행하기에 적합한 프락시를 선택한다.

IV. MLB(Mobile Load Balancing) Algorithm

프락시에 사용자가 요청한 비디오 스트림을 트랜스코딩 할 CPU 자원이 남아있지 경우 인접한 프락시에게 트랜스코딩 작업을 대신하여 수행하도록 해야 한다. 이때 트랜스코딩을 위한 적합한 프락시를 선택하기 위해 본 논문에서는 Mobile Load Balancing 알고리즘을 제안한다.

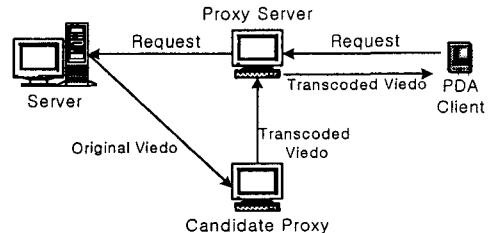
기존의 유선환경에서는 사용자가 이동하지 않기 때문에 로드밸런싱을 위한 프락시를 선택 할 때에 CPU의 사용량과 프락시에 존재하는 그룹 사용자의 수가 중요한 파라미터가 된다. 그러나 무선 환경에서는 해당 프락시 내에 존재하는 그룹 사용자가 다른 곳으로 이동할 가능성이 높기 때문에, 추가적으로 사용자의 이동성이 고려되어야 한다.

본 논문에서는 제안하는 MLB 알고리즘은 유선환경에서 고려되는 파라미터와 사용자가 과거동안에 이동한 경로 기록을 함께 반영하여 적용한다. 사용자의 경로 기록에서 공통적인 이동 패턴을 생성하기 위해서 Wen-Chin Peng et. al에서 이야기 된 log history기반의 사용자 이동 경로 패턴을 수정하여 무선 단말기 사용자가 처음 연결을 맺은 Home Agent에 자신을 등록하고 매 이동 시 자신의 단말기에 이동한 지역 정보를 가지고 있다가 연결이 종료될 때 Home Agent에 이동 경로를 보내어 주는 방법을 사용 한다.

MLB 알고리즘은 다음과 같다. 현재 CPU의 작업부하로 인하여 트랜스 코딩 작업을 수행할 수 없는 프락시를 기준으로 로드밸런싱의 후보가 될 수 있는 인접 프락시 그룹을 생성한다. 후보 프락시 그룹은 사용자에게 연속적으로 비디오 스트림을 제공할 수 있는 네트워크 지연 범위 내에 존재하는 기준으로 선택된다. 로드밸런싱을 요구 하는 프락시를 포함 하여 후보 프락시 그룹 내에 존재하는 모든 그룹 사용자의 이동 패턴을 생성한 후, 각각의 후보 프락시에 대하여, 가장 작은 값을 갖는 후보 프락시를 로드밸런싱을 하기 위한 프락시로 결정한다. 트랜스코딩을 수행한 프락시는 트랜스코딩 한 스트림을 요청한 프락시를 통해 사용자에게 제공하는 동시에 자신의 캐시 내에 저장하여 현재 동일한 비디오의 이전 부분을 보고 있는 다른 사용자(이하 그룹사용자라고 함)에게 가까운 미래에 제공한

다.

사용자 요청에 의한 MBL 알고리즘의 동작 과정은 [그림 2]와 같다.



[그림 2] MBL 알고리즘의 요청 과정

트랜스코딩 작업을 수행할 프락시([그림 2]에서 Candidate Proxy) 선정을 위한 MLB 값의 계산은 [수식 1]과 같다. 트랜스코딩을 수행할 프락시를 선정하기 위한 파라미터로는 CPU 사용률과 캐쉬 사용률, 네트워크 비용을 갖게 된다. [수식1]에서는 CPU 사용률과 캐쉬 사용률을 서로 더하고 그 값을 네트워크 비용을 나누어 가장 적은 값을 가지는 프락시를 선택하게 된다. [수식 1]에 나타난 네트워크 비용 계산은 [수식 2]와 같이 이루어진다.

[수식 2]의 COST_{NETWORK}는 후보 프락시에 트랜스코딩을 할 스트림을 캐싱하였을 때, 동일 비디오 요청 그룹의 이동성을 고려하여, 그룹 사용자들에게 해당 스트림을 제공하기 위한 네트워크의 비용을 계산하여 그 값이 가장 적은 프락시에게 우선순위를 부여하기 위해서 사용된다.

$$MLB_VALUE(j) = \frac{UTILITY_{CPU(j)} + UTILITY_{CACHE(j)}}{COST_{NETWORK(j)}}$$

[수식 1]

$$COST_{NETWORK(j)} = \frac{NM_{MovingGroupUser(j)}}{NM_{AllGroupUser}} * 100$$

[수식 2]

- $UTILITY_{CPU(j)}$: j 프락시의 현재 CPU 사용률
- $UTILITY_{CACHE(j)}$: j 프락시의 현재 CACHE 사용률
- $COST_{NETWORK(j)}$: j 프락시로 이동이 예상 되는 그룹 사용자의 비율
- $NUM_{AllGroupUser}$: 총 그룹 사용자의 수
- $NUM_{MovingGroupUser(j)}$: j 프락시로 이동하는 그룹 사용자의 수

V. 시뮬레이션

5.1 시뮬레이션 환경

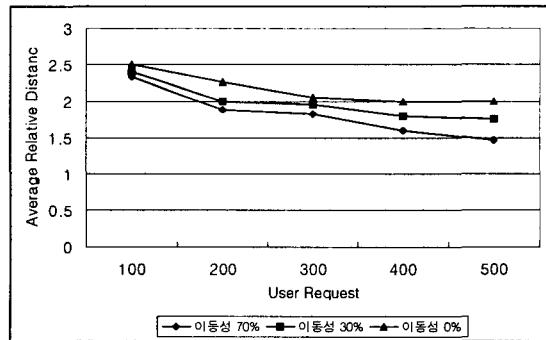
시뮬레이션을 위해 서버에서 제공하는 총비디오의 수는 100개로 가정하고, 각 비디오의 인기도는 Zipf-like 분포를 따르고, 사용자는 Poisson 분포에 따라 평균 20초의 간격으로 비디오 데이터를 요청한다고 가정하였다.

본 시뮬레이션에서 사용자는 하나의 비디오를 시청하면서 평균 5번의 위치 이동 기회를 가지며, 아래 [표 1]과 같이 서비스 받는 위치에 따른 상대적인 위치 값은 가진다.

| 서비스 경로 | 상대적인 위치 값 |
|-----------------------|-----------|
| 프락시 -> 사용자(cache hit) | 1 |
| 프락시-> 인접 프락시 -> 사용자 | 2 |
| 서버-> 프락시-> 사용자 | 4 |
| 서버 -> 인접 프락시 -> 사용자 | 6 |

[표 1] 서비스 경로에 따른 상대적인 거리

5.2 시뮬레이션 결과



[그림 3] Average Relative Distance

[그림 3]은 비디오 스트림을 요구하는 사용자를 일반적인 프락시 간 협동 기법을 이용한 방법과 본 논문에서 제안된 방법을 사용하였을 경우로 나누어 시뮬레이션 한 결과이다. 일반적인 협동 작업을 하는 프락시의 경우 사용자의 과거 이동 패턴에 대한 고려를 하지 않았으며, 제안된 방식을 사용할 경우 사용자가 과거 이동 패턴에 따라 이동할 확률을 30%와 70%로 각각 다른 비율을 주었을 경우를 시뮬레이션 하였다. 각 비디오 스트림은 서비스 받는 경로에 따라 [표 1]에서 정의한 상대적인 위치 값을 갖는다고 가정한다. 위의 그림과 같이 제안된 방식을 사용하여 과거 이동 패턴에

따른 이동할 확률을 30%로 정하였을 경우 평균값은 1.96로 나타났으며, 이동 패턴에 따라 이동 할 확률을 70%로 하였을 경우 1.82로 나타났다. 이 시뮬레이션의 결과 일반적인 협동 작업을 하는 프락시를 이용할 경우의 평균 값 2.16 보다 약 10% ~ 20% 네트워크 비용이 감소함을 확인하였다.

VI. Conclusion & Future work

본 논문에서는 무선 단말기 사용자에게 원활한 비디오 스트림 서비스를 제공하기위해 사용되는 트랜스코딩 작업시의 문제점인 CPU의 작업부하를 CPU의 사용률, 캐쉬의 여유 공간과 같은 유선환경에서 사용하던 파라미터에 사용자의 이동성 요소를 함께 고려하여 무선 환경에 적합한 로드밸런싱 방법을 제안하였으며, 시뮬레이션을 통하여 유선환경에서 사용되는 일반적인 프락시간의 협동 작업과 비교하여 성능이 개선됨을 보였다.

향후 무선 단말기 사용자에게 더욱 원활한 멀티미디어 스트림을 제공 하기위해 사용자의 이동 패턴을 좀 더 정확하게 예측할 수 있는 알고리즘 연구가 필요하다.

참고문헌

- [1] Xueyan Tang, Fan Zhang, Streaming Media Caching Algorithms for Transcoding Proxies, ICPP, 2002.
- [2] Wen-Chin Peng and Ming-Syan Chen, Developing data allocation schemes by incremental mining of user moving patterns in a mobile, IEEE Transactions on knowledge and data engineering, VOL. 15, No. 1, January/February 2003.
- [3] Bruce Zenel, A general purpose proxy filtering mechanism applied to the mobile environment, Wireless Networks 5, 391-409, 1999.
- [4] Mark Levene, Jose Borges, George Loizou, Zipf's Law for Web Surfers, Knowledge and Information Systems an International Journal, 3, 2001.