

VOD 시스템을 위한 P2P 프록시 기반의 효율적인 패칭 기법

권춘자*, 최치규*, 최황규*

*강원대학교 전기전자정보통신공학부

e-mail : kwoncj@mail.kangwon.ac.kr,

starflower22@mail.kangwon.ac.kr, hkchoi@kangwon.ac.kr

An Efficient Patching Scheme Based on P2P Proxy for VOD Systems

Chun-Ja Kwon*, Chi-Kyu Choi*, Hwang-Kyu Choi*

*Dept of Electrical and Computer Engineering, Kangwon National University

요 약

본 논문은 프록시의 prefix 패칭 기법을 적용하는 서버-프록시-클라이언트로 이루어진 VOD 시스템 상에서 서버에 대한 부하의 집중을 줄이기 위하여 클라이언트 시스템의 버퍼링과 재전송을 이용하는 P2P 프록시 패칭 기법을 제안한다. 제안된 기법은 패칭 윈도우의 크기를 벗어나는 요청에 대하여 VOD 서버가 새로운 정규 채널을 생성하는 대신 이전의 패칭 그룹에 속한 클라이언트로부터 정규 스트림을 전송받음으로써 서버로 집중되는 부하를 줄인다. 또한 프록시 서버를 기반으로 한 LAN 환경 내에서 클라이언트들의 상태 정보로 활용하기 위해 패칭 그룹에 대한 인덱스를 구성하였다. 시뮬레이션을 통해 prefix 크기와 비디오의 접근 패턴, 그리고 평균 요청 간격에 따른 서버의 대역폭 요구량을 측정하고 제안된 기법이 기존의 패칭 기법보다 서버의 대역폭 요구량이 현저히 감소함을 보인다.

1. 서론

온라인 교육이나 영화 등 사용자의 다양한 요구에 의하여 인터넷을 통한 미디어의 스트리밍 기술을 이용한 VOD(Video On Demand) 서비스의 수요가 점점 증가하고 있다. 대용량 멀티미디어의 실시간 스트리밍은 다수의 클라이언트들이 연속적인 전송을 요청하므로 많은 양의 네트워크 대역폭을 요구한다. 서버-클라이언트의 중앙집중식 VOD 시스템은 서버에 부하가 집중되므로 다수의 클라이언트들에 대한 서비스의 한계를 지닌다.

따라서 본 논문은 프록시 서버의 prefix 패칭 기법을 적용하는 서버-프록시-클라이언트로 이루어진 VOD 시스템 상에서 정규 채널을 생성하는 서버에 대한 부하의 집중을 줄이기 위하여 클라이언트 시스템의 버퍼링과 재전송을 통한 정규 스트림을 캐싱하는 P2P 프록시 패칭 기법을 제안하였다. 즉, 패칭 윈도우 크기를 벗어나는 요청에 대해 VOD 서버가 새로운 정규채널을 생성하는 대신 이전의 패칭 그룹에 속한 클라이언트로부터 정규 스트림을 전송받아 사용하도록 함으로써 서버의 부하를 줄인다. 이를 위하여 프록시 서버를 기반으로 한 LAN 환경 내에서 클라이언트를 패칭 윈도우 단위로 묶어 패칭 그룹에 대한 선형 구조로 구성하였다. 또, 프록시 서버 내에 클라이언트의 상태 정보를 유지하는 인덱스를 유지하도록 하여 불안정한 클라이언트 상에서 정규 스트

림 재전송시 클라이언트의 이탈로 인한 스트림 복구과정을 함께 제안하였다.

본 논문의 구성은 2장에서 관련 연구를 기술하고, 3장에서 제안된 기법의 P2P 프록시 패칭 기법에 대해 설명한다. 4장에서 시뮬레이션을 통해 제안된 기법의 성능분석을 수행하며, 마지막으로 5장에서 결론에 대해 기술한다.

2. 관련연구

VOD 시스템은 고성능의 서버라 할지라도 사용자의 증가에 따른 성능의 한계를 피할 수 없다. 따라서 서버로 집중되는 부하를 줄이기 위한 연구가 꾸준히 진행되어 왔으며, 주로 멀티캐스팅을 이용한 패칭 기법과 원거리의 서버로부터 일부의 데이터를 가져와 서버의 역할을 하는 프록시 서버를 활용하는 기법이 있다. 그리고 최근 들어 활발히 연구가 이루어지는 클라이언트 시스템을 활용하여 서버의 부하를 줄이는 P2P 미디어 스트리밍 기법이 있다.

멀티캐스트 패칭 기법은 멀티캐스팅 되는 정규 스트림의 버퍼링과 클라이언트 요청의 시간적인 차이에 따른 스트림을 동시에 전송받는다. 즉, 클라이언트의 요청이 들어오면 클라이언트 버퍼에 서비스 중인 스트림을 저장하는 동시에 지나간 prefix(앞부분) 스트림은 서버로부터 즉시 전송받아 재생하는 방식이다. 클라이언트들의 요청에 대하여 prefix 부분에 대한 스트림만을 전송하므로 서버의 부하를 크게 줄일 수 있다[1][2]. 그러나 클라이언트 버

본 논문은 2004년도 정보통신부 지원 기초기술 연구지원 사업 연구결과에의 일부임

퍼를 패칭 윈도우 크기로 하여 패칭 윈도우마다 새로운 정규 채널을 생성하여야 하므로 이것은 서버에 대한 부하로 작용한다.

프록시 캐싱 기법은 클라이언트와 네트워크 거리가 가까운 LAN 환경의 프록시 서버에 VOD 서버의 데이터를 캐싱하여 서비스함으로써 서버에 집중되는 부하를 분산하는 효과도 갖는다. 이와 같은 장점으로 프록시 서버에 비디오 파일의 prefix를 저장해 두어 초기 지연시간을 줄이는 Proxy Prefix Caching 기법이 제안되었다[3]. 또한 Proxy Prefix Caching 기법의 최적화를 통하여 네트워크의 대역폭 사용 비용을 최소화 하는 기법이 있다[4].

P2P 미디어 스트리밍 기법은 클라이언트 시스템을 통하여 스트림을 캐싱하므로 클라이언트 수가 증가함에 따라 전체 시스템 용량이 증가하는 장점이 있다. 그러나 불안정한 클라이언트 시스템을 캐싱 서버로 활용하므로 클라이언트의 이탈에 따른 복구과정이 필요하다. Chaining 기법은 다수의 클라이언트를 선형 구조로 연결한 후 버퍼링을 통하여 다른 클라이언트에 재전송하는 방식이다[5]. 기존의 패칭 기법과 P2P 미디어 스트리밍을 응용하여 prefix의 전송을 위하여 클라이언트로부터 캐싱을 수행하는 P2CAST 기법도 제안되었다[6]. 그러나 서버가 정규 채널을 주기적으로 생성하므로 정규 스트림에 대한 부하가 요구된다.

3. P2P 프록시 패칭 기법

3.1 프록시 서버를 이용한 P2P 패칭 시스템

본 논문에서 제안된 시스템 구성은 그림 1과 같이 인기도가 높은 비디오 파일의 앞부분(prefix)을 저장하는 프록시 서버와 그 뒷부분(suffix)을 제공하는 VOD 서버, 그리고 자신과 가장 가까운 프록시 서버에 연결되어 버퍼링을 수행하는 다수의 클라이언트 시스템으로 구성된다.

하나의 프록시 서버에 연결된 다수의 클라이언트들이 특정 비디오에 대한 요청을 수행할 때, 각 요청에 대해 패칭 윈도우 단위로 클라이언트 그룹이 형성된다. 이렇게 패칭 그룹이 연속적으로 생성되는 경우 기존의 패칭 기법은 패칭 윈도우 크기마다 정규채널을 생성하는 것에 반하여, 제안된 기법은 클라이언트의 버퍼링과 지연전송을 이용하고 그룹 간 캐싱에 의해 연속적인 패칭 그룹이 지속되는 동안 서버로부터 정규 채널을 생성하지 않으며 대신 이전의 패칭 그룹에 속한 클라이언트에 버퍼링된 스트림을 전송받으므로 빈번한 정규 채널의 생성으로 인한 서버의 부하를 줄일 수 있다.

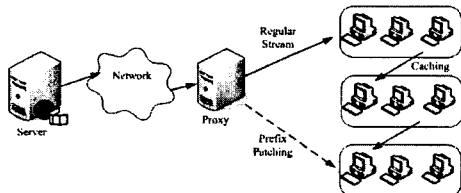


그림 1. P2P 프록시 패칭 기법의 시스템 구성

본 시스템에서 안정적인 서버가 아닌 불규칙한 클라이언트로부터 스트림을 제공받는 것이므로 클라이언트의 이탈에 따른 복구 방법과 동작으로 움직이는 다수의 클라이언트에 대한 정보의 관리가 필요하다. 따라서 prefix 패칭 스트

림을 제공하고 패칭 그룹에 대한 인덱스 정보를 제공하는 프록시 서버의 역할과 서버로서의 기능을 수행하는 클라이언트들의 동작을 설명한다. 클라이언트들의 동작에 따른 관리는 새로운 클라이언트의 참여 방법, 클라이언트의 버퍼링을 통한 재전송 과정, 그리고 클라이언트의 불규칙한 행동에 의한 이탈에 따른 복구 방법 등으로 이루어지며 상세한 수행과정은 다음과 같다.

3.2 프록시 서버의 역할

VOD 시스템에서 프록시 서버는 클라이언트와의 세션에 대한 정보를 유지하고 상태에 대한 변화를 갱신하는 역할을 수행하여야 한다. 따라서 불안정한 클라이언트 시스템을 정규 스트림의 패칭 서버로 이용하는 본 논문에서는 클라이언트의 불규칙적인 행동으로 인한 대체 및 복구 과정을 수행하기 위해 프록시 서버에 유지되는 인덱스 정보를 이용한다. 즉, 프록시 서버는 prefix의 패칭 스트림을 제공하는 역할과 더불어 인덱스 서버로서의 역할도 함께 수행한다.

그림 2는 연결된 모든 클라이언트들에 대한 세션 정보를 유지관리하기 위한 프록시상의 인덱스 구조를 나타낸 것이다. 그림에서 prefix 크기를 단위로 하는 그룹들 중 G1은 서버로부터 전송된 정규 스트림이며 G2의 정규 스트림은 G1에 속하는 클라이언트의 버퍼에 저장되었다가 전송된다. 이 경우 G2의 클라이언트는 프록시 서버로부터 prefix를 패칭한다. 또한 G3은 이전 그룹 G2와의 시간적인 차이가 prefix 크기보다 커 캐싱할 수 있는 범위를 벗어나므로 서버로부터 다시 정규 스트림을 전송받는다. 또한 G3과 G4간의 시간간격은 prefix 크기보다 작으므로 G4는 서버로부터 추가적인 정규 채널을 요구하지 않고 G3의 클라이언트로부터 캐싱된 정규스트림을 전송받는다. 구성된 인덱스에는 각 그룹의 시작시간과 재생중인 클라이언트 정보, 각 그룹을 위한 정규 스트림이 서버로부터 전송되는 것인지 이전 클라이언트의 그룹에 의해 캐싱되는 것인지에 대한 정보도 포함한다.

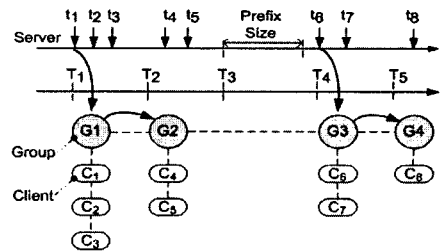


그림 2. P2P 프록시 패칭을 위한 인덱스 구성

3.3 클라이언트들의 동작 관리

1) 새로운 클라이언트의 참여

제안된 기법에서 프록시 서버에서 패칭되는 prefix 크기와 클라이언트 시스템에 버퍼링되는 정규 스트림의 크기는 동일하다.

만일 새로운 클라이언트가 특정 비디오에 대한 요청을 하는 경우 가장 최근에 생성된 그룹의 시작 시간과 현재 클라이언트의 요청 시간을 비교하여 정규 채널을 사용하는 새로운 그룹을 생성할 것인지, 혹은 최근의 그룹으로부터 캐싱된 정규 채널을 재전송 받는 그룹을 생성할 것인지, 아니면 기존의 그룹에 참여할 것인지 결정하여야 한다. 즉, 최근 그룹의 시작시간과 클라이언트의 요구 시간의 차가 prefix의 크

기보다 작다면 기존 그룹에 참여하는 과정을 수행하고, 반대로 최근 그룹의 시작 시간과 현재의 요청 시간의 차가 prefix 크기보다 크다면 새로운 패칭 그룹을 생성하여 클라이언트를 참여시킨다. 이렇게 결정되는 패칭 그룹과 클라이언트의 관계는 프록시 서버 상에 유지되는 인덱스 정보와 일치한다.

2) 클라이언트의 버퍼링과 재전송 과정

프록시를 이용한 prefix 패칭 시스템 상에서 클라이언트는 정규 스트림을 버퍼링하여 이후의 그룹에 지연전송을 수행한다. 즉, 기존의 패칭 기법이 재생을 위하여 정규 스트림과 요청 시간의 차에 해당하는 분량만을 버퍼링하는 것에 비하여 제안된 기법은 재생을 위한 버퍼링과 동시에 재전송을 위한 버퍼링을 함께 수행한다.

그림 3은 각각의 패칭 그룹과 그에 속한 클라이언트들에 의하여 연속되는 그룹들 간의 정규 스트림이 캐싱되는 과정을 보여준다. 여기서 하나의 그룹은 하나 이상의 클라이언트를 가지며 하나의 클라이언트 시스템은 이어지는 그룹에게 멀티캐스트를 통하여 정규스트림을 재전송하는 캐싱 서버의 역할을 한다. 또 같은 그룹에 속한 클라이언트라 하더라도 서로의 요청시간에 따라 재생 지점과 전송을 위한 버퍼와 재생을 위한 버퍼 공간의 비율이 다르게 된다.

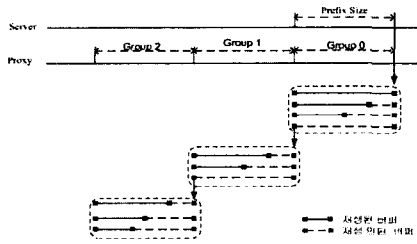


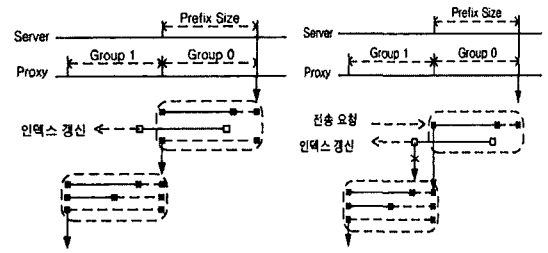
그림 3. 패칭 그룹에 의한 정규 스트림 캐싱

3) 클라이언트 이탈에 따른 복구 과정

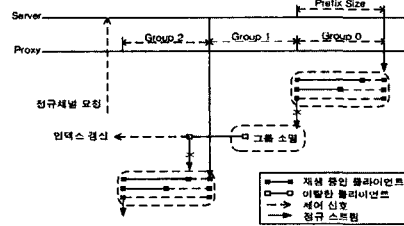
제안된 기법은 클라이언트 시스템을 정규 스트림에 대한 캐싱 서버로 활용하므로 안정적인 서버와는 달리 클라이언트의 불규칙한 행동이나 재생이 종료되어 멀티캐스팅 중인 클라이언트가 그룹에서 이탈하는 경우가 발생한다. 따라서 전송중인 클라이언트가 그룹에서 이탈하는 경우 이탈에 따른 복구 과정은 프록시 서버에 유지되는 인덱스를 이용해서 이루어진다.

그림 4는 클라이언트의 이탈에 대한 정규 스트림의 복구 과정을 나타낸다. 그림 4의 a)는 재전송을 수행하지 않던 클라이언트가 그룹에서 이탈을 원하는 경우이며, 프록시에 이탈에 대한 신호를 보내면 프록시는 이에 대한 인덱스를 갱신하여 클라이언트의 상태와 인덱스의 내용을 동일하게 유지시킨다. 또 그림 4의 b)는 재전송을 수행 중인 클라이언트가 이탈하는 경우에 대하여 그룹 내의 새로운 클라이언트를 선택하여 스트림의 전송을 대체하는 경우를 보여준다. 이때 프록시 서버는 이탈을 요청한 클라이언트의 신호에 대하여 인덱스를 갱신하고, 동시에 스트림의 전송을 대체할 클라이언트를 검색하여 스트림을 전송할 것을 요청한다. 마지막으로 그림 4의 c)는 그룹 내의 모든 클라이언트가 이탈한 경우를 나타낸다. 이탈을 희망하는 클라이언트가 프록시에 이탈을 요청하면 프록시는 이탈에 대한 인덱스를 수정한다. 이때 해당하는 그룹의 클라이언트가 모두 나갔으므로 대체할 클라이언트가 없다. 이런 경우 프록시는 VOD 서버에게 비디오

의 남은 분량에 대한 정규 스트림을 요청한다.



a) 재전송을 수행하지 않는 클라이언트 이탈 b) 재전송 수행 중인 클라이언트 이탈



이 클라이언트의 이탈로 인한 그룹의 소멸

그림 4. 클라이언트의 이탈과 복구 과정

4. 성능평가

제안된 기법의 성능 평가를 위하여 표1에 정의된 파라미터를 사용하여 기존의 패칭 기법과 제안된 기법에 대한 시뮬레이션 수행하였다. 평가 모델에서 각각의 프록시는 인기도가 높은 상위 10%의 비디오의 prefix 부분을 저장한다. 또, VOD 서버에 대한 클라이언트의 요청은 λ 의 발생 빈도를 갖는 포아송 분포를 따른다고 가정하며, 전체 비디오에 대한 요청 패턴이 Zipf-like 분포를 따른다고 가정할 때, 비디오 i 에 대한 요청 확률은 $P_i = \frac{C}{i^{1-\theta}}$ 의 값을 갖는다[7].

표 1. 성능 평가를 위한 파라미터

parameter	default	variation
비디오 수	100	N/A
비디오의 길이(minutes)	90분	N/A
Mean-Inter Arrival Time (seconds)	$1/\lambda$	10
Prefix 크기(minutes)	10분	0-20
Normal Playback Rate b(Mbps)	1.5	N/A
Skew Factor	0.271	0.0-1.0
프록시 서버의 수	7	N/A
시뮬레이션 시간(hours)	10	N/A

시뮬레이션 결과에서 서버의 대역폭 요구량은 10시간 분량의 요청에 대하여 서버가 사용한 대역폭의 전체 소모량을 구하였으며, 5회 반복한 결과의 평균으로 최종 수치를 구하였다.

그림 5는 프록시 서버에 저장된 prefix의 크기에 따른 서버의 대역폭 요구량을 나타낸다. 전체 100개의 비디오들 중에서 인기도가 높은 10%의 비디오만이 프록시에서

서비스 된다고 가정하였고, 평균 요청 간격은 10초이다. 또한 프록시에 저장된 prefix의 크기는 패칭 윈도우의 크기와 일치한다. 기존의 패칭 기법과 비교하여 볼 때 제안된 클라이언트 시스템을 이용한 정규 스트림의 재전송을 통한 캐싱 기법이 prefix 크기의 증가에 따른 평균 대역폭의 요구량이 훨씬 적게 나타났다.

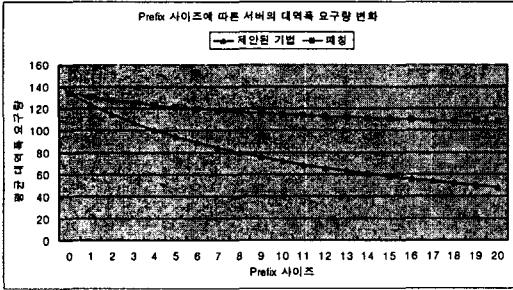


그림 5. Prefix의 크기에 따른 서버 대역폭 요구량

그림 6은 클라이언트의 평균 요청 간격에 따른 서버의 대역폭 요구량을 보인다. 이때 인기도가 높은 상위 10%의 비디오에 대해 프록시 서버에 저장된 prefix 크기는 10분으로 하였다. 평균 요청 간격이 커짐에 따라 서버의 대역폭 요구량이 줄어드는 것을 볼 수 있다. 또한 제안한 기법이 기존의 패칭 기법보다 동일한 요청 간격에서 서버에 대한 대역폭 요구량이 작게 나타나며, 특히 요청 빈도가 높을수록 제안된 기법의 효과가 크게 나타남을 볼 수 있다.

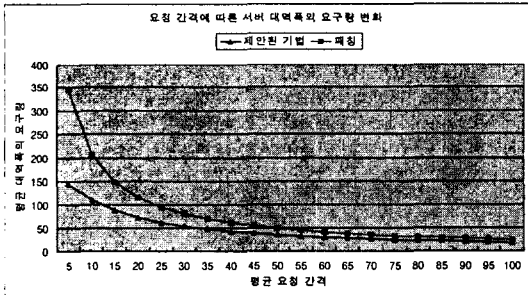


그림 6. 평균 요청 간격에 따른 서버 대역폭 요구량

그림 7은 전체 비디오에 대한 요청 패턴인 θ 의 변화에 따른 대역폭 요구량을 나타낸다. 이때 θ 의 범위는 0.0에서 1.0이며 θ 의 값이 작을수록 특정 비디오에 대한 요청비율이 높은 것을 나타낸다. 제안된 기법이 기존의 패칭 기법보다 동일한 θ 값에서 서버의 대역폭을 훨씬 작게 요구하는 것을 볼 수 있다.

5. 결론

본 논문은 기존의 중앙 집중 방식의 서버-클라이언트 시스템과는 달리 클라이언트 시스템에 저장된 정규스트림의 캐싱을 사용하는 P2P 프록시 패칭 기법을 제안하였다. 제안 기법은 패칭 윈도우 크기의 시간을 벗어나는 클라이언트의 요청을 위해 서버에 새로운 정규 채널을 생성하는 대신 이전의 패칭 윈도우에 속한 클라이언트 시스템에 버퍼링된 스트림을 재전송 받아 사용하는 방식이다. 이를 위하여 프록시

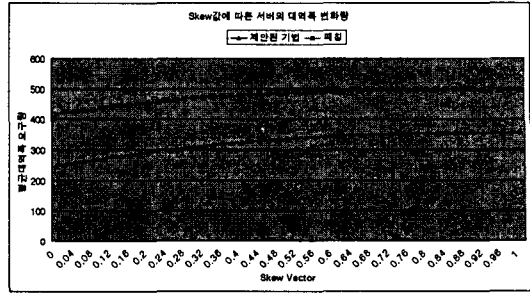


그림 7. Skew factor 값에 따른 서버 대역폭 요구량

서버를 prefix를 제공하는 패칭 서버로 활용하는 동시에 클라이언트의 상태를 기록하고 탐색하기 위한 인덱스 서버로 활용하였다. 시뮬레이션을 통해 제안 기법의 성능 평가를 위하여 동일 조건하에서 프록시의 prefix 크기, 비디오에 대한 평균 요청 간격 그리고 비디오 요청 패턴에 따른 대역폭의 요구량은 제안된 기법이 기존의 패칭 기법에 비하여 크게 감소함을 보였다.

참고 문헌

- [1] K. A. Hua, Y. Cai, and S. Sheu, "Patching: A Multicast Technique for True Video-on-Demand Services", In Proc. of ACM Multimedia '98, Bristol, U.K., Sep. 1998.
- [2] Y. Cai, K. A. Hua and K. Vu, "Optimizing Patching Performance", In Proc. of SPIE's Conference on Multimedia Computing and Networking '99, San Jose, Jan. 1999.
- [3] S. Sen, J. Rexford, and D. Towsley, "Proxy Prefix caching for Multimedia Streams", In Proc. of the IEEE Infocom, Vol. 3, 1998.
- [4] B. Wang, S. Sen, M. Adler, and D. Towsley, "Optimal Proxy Cache Allocation for Efficient Streaming Media Distribution", In Proc. of the IEEE Infocom, Vol. 3, New York, NY, June 2002.
- [5] S. Sheu, K. A. Hua, and W. Tavanapong, "Chaining: A Generalized Batching Technique for Video-On-Demand Systems", In Proc. Of IEEE Int'l Conf. On Multimedia Computing and Systems (ICMCS'97), pp. 110-117, Ottawa, Canada, June 1997.
- [6] Y. Guo, K. Suh, J. Kurose, D. Towsley "P2Cast: Peer-to-peer Patching Scheme for VoD Service", Proceedings of the 12th World Wide Web Conference (WWW-03), Budapest, Hungary, May 2003.
- [7] A. Dan, D. Sitaram, and P. Shahabuddin. "Dynamic batching policies for an on-demand video server", Multimedia Systems, 4(3):112-121, June 1996.