

모바일 환경에서의 협력작업을 하는 프록시 서버를 위한 캐쉬 교체정책

장해권⁰ 한중현 정홍기 박승규
아주대학교 정보통신전문대학원
{crjang95⁰, hanbell, keeper, sparky}@ajou.ac.kr

The Cache Replacement Policy for Collaborative Proxy Servers in Mobile Environments

Haegwon Jang⁰ Jonghyun Han Hongki Jung Seungkyu Park
Graduate School of Information and Communication AJOU University

요 약

무선환경과 모바일 기기의 발달되고, 모바일 단말기의 보편화로 인해 유선환경에서 제공되던 각종 멀티미디어 서비스가 무선환경으로 옮겨가고 있다. 기존의 웹캐쉬 정책은 그 성능을 이미 검증 받았다. 그러나 그런 정책들은 인기도와 네트워크의 상태만을 고려했기 때문에 무선환경에서는 적합하지 않다. 본 논문에서는 무선환경에서 모바일 호스트의 이동성과 미디어 스트림의 특성을 고려한 캐쉬 교체 정책인 M-LRU를 제안한다. 그리고, 시뮬레이션을 통하여 기존의 LRU정책과 제안한 M-LRU 정책을 비교 하였으며, 8-9%의 성능향상이 되었다는 것을 보여준다.

1. 서 론

무선 네트워크의 발달과 모바일 단말기의 보급이 보편화 되면서 기존의 유선환경에서 제공되던 각종 서비스들이 무선환경으로 옮겨가고 있다. 그러나 오브젝트의 크기가 큰 멀티미디어 스트림을 특성과 무선 환경의 특성인 낮은 대역폭, 작은 저장용량, 이동성등으로 서비스 제공의 어려움이 있다.

기존의 웹캐쉬는 사용자들의 요청빈도가 높은 스트림을 가까운 로컬 캐쉬 서버에 저장함으로써 보다 신속하고 일정한 응답시간을 제공함으로써 사용자에게 대한 서비스 효율과 품질 개선으로 그 성능이 검증되었다. 웹캐쉬의 성능은 캐쉬 교체 정책에 따른 Hit Ratio에 크게 의존한다.

그러나 무선환경에서는 기존의 웹 캐쉬정책이 적합하지 않다. 기존의 웹 캐쉬 교체 정책들은 멀티미디어 스트림의 크기와 네트워크의 상태, 그리고 인기도에 따라 적용되는 정책들이다. 그러나 무선환경에서는 이러한 요소 뿐만아니라 모바일 호스트의 이동성을 고려해야한다.

본 논문에서는 기존의 교체정책인 LRU를 무선환경에서 적합하게 확장한 M-LRU 정책을 제안한다. 본 논문에서 제안하는 정책은 미디어 스트림의 특성과 모바일 호스트의 이동성을 고려, 인기도와 함께 로컬 캐쉬의 교체를 결정하는 알고리즘이다.

본 논문의 구성은 2장에서는 관련연구를 살펴보고, 3장에서는 제안하는 캐쉬 교체 알고리즘을 설명하고, 4장에서는 제안한 알고리즘과 기존의 알고리즘을 시뮬레이션을 통하여 비교분석 하였으며, 5장에서는 결론 및 향후계획에 대해서 기술한다.

2. 관련연구

기존의 웹 캐쉬 정책으로 가장 잘 알려진 알고리즘은 LRU와 LFU이다. O'Neil이 제안한 LRU-K 알고리즘은 오브젝트의 지난 K번에 대한 접근 시간 정보를 가지고 있다. 이 정보를 바탕으로 오브젝트의 교체를 결정한다. Young이 제안한 Greed-Dual 알고리즘은 동일한 크기를 같은 오브젝트들을 캐쉬로 가져올 때의 비용이 다른 경우에 적용하기 위해 제안 되었는데, 각 오브젝트들이 우선순위(H)를 가지고 있고, 그 값이 가장 작은 오브젝트를 교체한다. 이때 캐쉬 되어있는 다른 오브젝트의 H값에서 교체되는 오브젝트의 H값만큼을 감소 시킨다. 그리고 만약 캐쉬 히트 될경우에 원래의 H값으로 복원시킨다. [1]

이러한 알고리즘들은 웹 캐쉬를 기반으로 한 알고리즘들이어서 미디어 스트림과 같은 크기가 크고, 사용자의 요구에 대한 반응이 민감한 멀티미디어 데이터에는 적합하지 않다. 그래서 제안된 알고리즘이 prefix 캐쉬이다. prefix 캐쉬는 미디어 스트림의 앞부분은 프록시에서 가지고 있고, 나머지 부분은 비디오 서버에서 가지고 있다. 사용자 요청시 가까운 곳에 위치하고 있는 프록시로부터

본 연구는 정보통신부지원 대학기초연구지원사업(정보통신기초기술연구지원사업)의 지원을 받아 진행됨.

prefix부분을 먼저 서비스 받고, 동시에 프록시가 서버에게 나머지 부분을 요청하여 서버로부터 받게 됨으로 해서 초기응답 시간을 최소화 하는 방법이다. [2]

그러나 이러한 캐쉬 정책들은 유선환경을 기반으로 하고 있기 때문에 이동성이 강한 무선환경에서는 적합하지 않다.

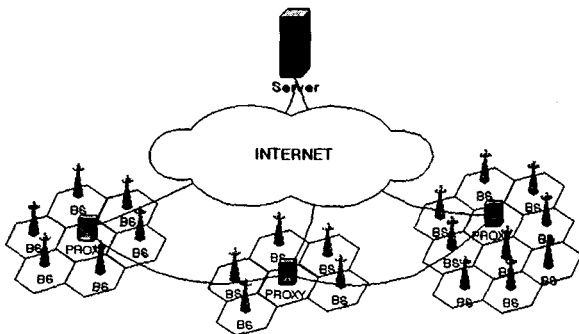
본 논문에서는 무선환경에서 모바일 호스트의 이동성과 미디어 스트림의 특성에 적합한 알고리즘을 제안한다.

3. 캐쉬 교체 알고리즘

앞에서 살펴본 기존의 알고리즘들은 무선환경에서 사용하기에는 적합하지 않다. 본 논문에서는 무선환경에 적합하도록 LRU 캐쉬정책을 확장하려고 한다.

3.1 시스템 구성

<그림 1>은 멀티미디어 스트림을 제공하기 위한 시스템이다. 모든 멀티미디어 스트림은 멀티미디어 서버에서 저장하여 가지고 있고, 각 멀티미디어 스트림은 세그먼트 형태로 나누어져있다. 서버는 프록시가 중계하는 사용자의 요구에 따라 미디어 스트림을 프록시로 전송한다.



<그림 1> 시스템 구성

프록시는 기본적으로 각 비디오 세그먼트의 prefix를 가지고 있고, 사용자가 비디오를 요청하였을때, prefix를 먼저 서비스하고 나머지 suffix 부분을 서버와 인근 프록시로 요청하여 서비스하는 형태이다. 따라서 교체정책은 suffix부분을 대상으로 한다.

근접해 있는 프록시들간에는 협동작업을 통하여 미디어 스트림을 제공한다. 만약 새로운 사용자 요청이 들어왔을때 현재 프록시에 요청한 미디어 스트림이 없다면, 서버와 프록시들에게 미디어 스트림에 대한 요청을 보내서 가장 빠른 응답을 보이는 곳으로부터 미디어 스트림을 전송받아서 서비스 한다.

프록시에 저장된 데이터가 미디어 데이터라는 것을 고려하여, 본 논문에서는 semilock과 unlock의 두 단계의 캐쉬 구조를 가지고 있다. 현재 프록시의 로컬 영역이나 프록시가 전송해 줄 수 있는 범위 내에서 사용자의 요청이 들어오면 해당 미디어 스트림은 semilock 상태가 되

는데 semilock은 미디어 스트림이 다른 프록시에게 세그먼트를 보내 주어야 하기 때문에 프록시에서 캐쉬 교체가 이루어 질때 무조건 교체되는 것을 막기 위해서 일종의 lock과 같은 역할을 한다. 이와 반대로 서비스 하고 있지 않은 미디어 스트림은 unlock 상태가 된다. 캐쉬 교체가 되어야 할 경우 먼저 unlock 상태의 미디어 스트림을 대상으로 교체가 되고, unlock 상태에 있는 미디어 스트림이 없는 경우에는 semilock 상태의 미디어 스트림을 교체 하게된다. 이때 교체 정책은 M-LRU 정책을 사용한다.

3.2. M-LRU

모바일 호스트가 서비스 받는 지역을 옮기더라도 최대한 캐쉬의 Hit ratio를 높이기 위해 알고리즘을 제안한다.

$$P_i = \alpha \times \beta_i \times C_i \quad \text{<수식 1>}$$

α : 프록시간 미디어 요청시 네트워크 딜레이를 고려한 가중치로 프록시가 서비스하는 로컬영역에서 서비스가 이루어지는 경우 1, 그렇지 않고 리모트 프록시에서 서비스를 요청하는 경우 딜레이 시간을 고려 $f(t)$ 를 계산하여 미디어 스트림이 있는 프록시로 α 를 보내준다.

$$\alpha = f_a(\text{delay}) \text{ or } \alpha = 1 \quad (0 < \alpha \leq 1) \quad \text{<수식 2>}$$

β : semilock 되어있는 스트림의 세그먼트들 중에서 현재 서비스 되고있는 세그먼트로부터의 거리.

$$\beta = f_b(\text{seg}_i) \quad (0 < \beta \leq 1) \quad \text{<수식 3>}$$

C : 현재 미디어 스트림을 요청한 모바일 호스트의 수로 로컬영역에서의 서비스를 받는 모바일호스트와 리모트에서 서비스를 요청하는 모바일호스트의 수의 합이다.

교체될 미디어 스트림이 선택되면, 미디어 스트림의 연속성을 보장하기위해 마지막 세그먼트의 suffix를 교체한다.

프록시에 원하는 suffix가 없다면, 프록시는 인근 프록시와 서버로 스트림을 요청하는 방송을 한다. 방송을 하여 응답이 오는 시간을 측정하여, 인근 프록시 마다 α 계산후 계산된 α 를 해당 프록시에 알려준다. 이때 suffix가 없거나 있다하더라도 딜레이가 커서 비디오 스트림의 전송과 재생이 동시에 이루어지더라도 서비스가 제대로 이루어지지 않을 프록시에는 관심을 갖지 않는다. 그리고 α 값을 받은 프록시에서는 로컬영역에 서비스를 제공하고 있다면 α 값을 1로 그렇지 않다면, 전송받은 α 값으로 멀티미디어 스트림의 우선순위 P_i 를 계산한다. 전송받은 α 가 여러개일 경우 최대값을 취한다.

사용자가 서비스 받는 연속된 세그먼트는 semilock 상태에 있다. 사용자가 하나의 세그먼트를 다 본 경우에 그 세그먼트는 unlock 상태가 된다. 이 세그먼트는 자신이 unlock 상태로 바뀐 시간 정보를 가지고 있고, 세그먼트는 미디어 스트림 별로 관리한다. 이때의 캐쉬 교체 정책은 각 스트림의 세그먼트 정보를 모아서 그중에서

가장 오래된 미디어 스트림의 가장 뒷부분부터 교체하게 된다. unlock 상태의 세그먼트중에서 다른 사용자의 요청을 받으면 다시 semilock 상태로 상태가 바뀌게 된다.

unlock 상태의 세그먼트가 하나도 없다면, 그때는 semilock 상태의 미디어 스트림에서 교체하게 된다. 현재 프록시 내에서 교체 대상을 찾을때에는 P_i 값을 계산하여 가장 작은 값의 세그먼트를 교체하게 된다.

4. 시뮬레이션

4.1. 시뮬레이션 환경

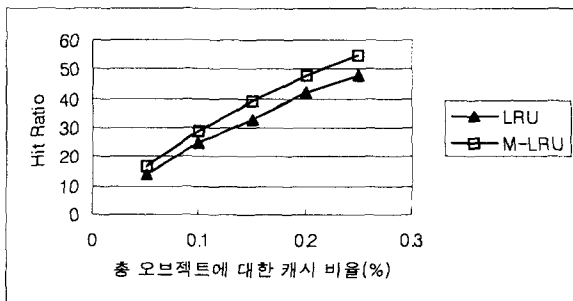
서버에 위치한 전체비디오의 수는 100개이고, 비디오의 길이는 동일하며, 각 비디오는 100개의 세그먼트로 분할되어있다. 프록시들은 각 세그먼트의 prefix 부분을 가지고 있다. 서버는 하나만 존재하고 프록시 서버는 5개가 있다고 가정하였다.

각 미디어 스트림의 인기도는 Zipf-like 분포를 따르고, 사용자의 수는 1000명으로 Poisson 분포(평균 타임 인터벌 20초)를 사용하였다.

그리고 가장 많이 사용하는 웹 캐쉬 알고리즘인 LRU와 본 논문에서 제안하는 M-LRU를 시뮬레이션을 통하여 비교하였다. 시뮬레이션에서 측정하는 파라미터는 캐시의 성능을 대표할 수 있는 Hit-ratio와 Byte Hit-ratio를 사용한다.

4.2. 시뮬레이션 결과

<그림 2>는 총 오브젝트에 대한 캐시 비율에 따른 LRU와 M-LRU의 캐쉬 Hit-Ratio를 측정한 결과를 보여준다.

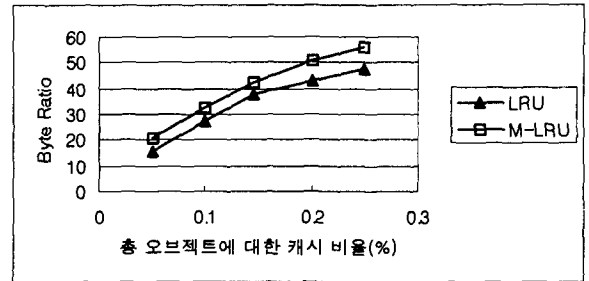


<그림 2> M-LRU의 Hit-Ratio

<그림 2>에서 보여지듯이 Hit-Ratio는 M-LRU는 기존의 캐쉬 교체 정책인 LRU에 비해서 프록시의 캐쉬 크기가 커질수록 성능향상이 되었으며, 캐쉬의 크기가 전체 오브젝트 크기의 25%정도일때 8%의 성능 향상을 보여 준다.

<그림 3>은 총 오브젝트에 대한 캐시 비율 변화에 따른 LRU와 M-LRU의 Byte Hit-Ratio를 측정한 결과를 보여준다. <그림 3>에서 보여 지듯이 Byte Hit-Ratio 또한 LRU보다 M-LRU가 캐쉬의 크기가 커질수록 성능

이 향상 되었으며, 캐쉬의 크기가 전체 오브젝트의 25% 일때 9% 정도의 성능향상이 있었다.



<그림 3> M-LRU의 Byte Hit-Ratio

5. 결 론 및 향후과제

본문에서는 무선환경에서 움직이는 모바일 호스트들에게 멀티미디어 서비스를 제공함에 있어서 중요한 역할을 하는 프록시의 캐쉬 교체정책을 제안하였다. 기존의 멀티미디어 서비스를 캐쉬하는 프록시들의 캐쉬 교체정책을 무선환경에서 이동성을 가지는 호스트들이 서비스를 좀더 원활하게 받을 수 있도록 개선하였으며, 제안된 방법은 캐쉬의 크기가 커질수록 더욱 좋은 효율을 보여 준다.

향후 과제로는 단순히 모바일 호스트의 이동성만이 아닌 사용자의 이동 패턴 분석을 통하여 방향성과 이동속도까지도 감안한 캐쉬 교체정책이 연구중에 있다.

6. 참고문헌

- [1] Martin Arlitt, Rich Friedrich, and Tai Jin, "Performance Evaluation of Web Proxy Cache Replacement Policies" Hewlett-Packard Laboratories, 1501 Page Mill Road, Palo Alto, CA 94304
- [2] S. Sen, J. Rexford, and D. Towsley, "Proxy prefix caching for multimedia streams," in *Proc. IEEE INFOCOM*, April 1999.
- [3] P. Lorensetti, L. Rizzo and L. Vicisano, "Replacement Policies for Proxy Cache", Manuscript, 1997.
- [4] Kun-Lung Wu, Philip S. Yu and Joel L. Wolf, "Segment-Based Proxy Caching of Multimedia Streams", WWW10, ACM, May 1-5 2001
- [5] Kevin W. Froese and Richard B. Bunt, "Cache Management for Mobile File Service", *The COMPUTER JOURNAL* Vol. 42, No. 6, 1999
- [6] Bing Wang, Subhabrata Sen, Micah Adler and Don Towsley, "Optimal Proxy Cache Allocation for Efficient Streaming Media Distribution", *INFORCOM* 2002