

비디오 프록시 서버에서의 시간 제약 다중 요청 기법 기반 동영상 데이터 관리

이준표*, 조철영*, 권철희*, 이종순*, 김태영*

Video Data Management based on Time Constraint Multiple Access Technique in Video Proxy Server

Jun Pyo Lee*, Chul Young Cho*, Cheol Hee Kwon*, Jong Soon Lee*, Tae Yeong Kim*

요약

본 논문에서는 비디오 프록시 서버의 제한된 저장 공간을 효율적으로 활용하기 위한 시간 제약 다중 요청 기법을 제안한다. 제안하는 기법은 요청된 동영상 데이터를 전송받아 사용자에게 전송하고 비디오 프록시 서버에 일시적으로 저장한다. 이때 일시적으로 저장된 동영상 데이터는 설정된 시간 내에서 발생하는 사용자의 요청의 상태에 따라 저장장치에서 삭제되거나 저장된다. 또한 새롭게 요청된 동영상의 저장 공간을 확보하기 위해서 저장장치에 저장되어 있는 동영상 세그먼트 중 요청 가능성이 가장 낮은 세그먼트를 선정하고 제거한다. 이를 위해 사용자에게 의해 주로 요청되는 동영상 세그먼트 부분인 전방 클래스와 요청되지 않았거나 요청될 가능성이 적은 세그먼트 부분인 후방 클래스로 분리한다. 분리된 클래스 중 후방 클래스에서 가장 오래전에 요청된 세그먼트를 선정하여 삭제함으로써 제한된 공간을 효율적으로 활용한다. 실험을 통해 제안하는 방법이 기존의 방법들 보다 높은 적중률을 보이는 동시에 보다 적은 삭제 횟수를 보인다는 것을 확인한다.

Abstract

Video proxy server which is located near clients can store the frequently requested video data in storage space in order to minimize initial latency and network traffic significantly. However, due to the limited storage space in video proxy server, an appropriate video selection method is needed to store the videos which are frequently requested by users. Thus, we present a time constraint multiple access technique to efficiently store the video in video proxy server. If the video is requested by user, it is temporarily stored during the predefined interval and then, delivered to the user. A video which is stored is deleted or moved into the storage space of video proxy server depending on the request condition. In addition, we propose a video deletion method in video proxy server for newly stored video data. The simulation results show that the proposed method performs better than other methods in terms of the block hit rate and the number of block deletion.

• 제1저자 : 이준표
• 투고일 : 2010. 09. 10, 심사일 : 2010. 09. 20, 게재확정일 : 2010. 09. 27.
* LIG 넥스원 지휘통제연구센터

▶ Keyword : 비디오 프록시 서버(video proxy server), 비디오 스트리밍(video streaming), 스트리밍 서비스(streaming service), 비디오 저장 기법(video storing technique)

I. 서론

사용자들과 근거리에서 위치하는 비디오 프록시 서버(video proxy server)는 사용자들에 의해 요청되어지는 대용량의 동영상 데이터들을 복잡한 인터넷의 중간 경로를 거치지 않고 다수의 사용자들에게 직접 전달함으로써 데이터의 손실과 초기 전송 지연(initial latency)문제를 해결한다[1]-[6]. 그러나 비디오 프록시 서버는 주 서버(central video server)의 저장 공간에 비해 비교적 제한된 용량을 가진다는 단점이 있다. 따라서 사용자에 의해 앞으로 사용자에 의해 계속 요청될 동영상 데이터만을 저장하도록 하는 방법이 필요하다. 이를 위한 연구에서 사용자가 주로 동영상의 시작 부분을 요청하는 접근 패턴을 가지며 비교적 짧은 지속시간을 가진다는 점을 활용하여 사용자가 요청한 동영상의 시작 부분인 프리픽스(prefix)만을 비디오 프록시 서버에 저장 하는 프리픽스 저장(prefix caching) 기법이 제안되었다[7]. 프리픽스 저장 기법은 사용자가 동영상을 요청했을 때 비디오 프록시 서버에 저장 되어있는 해당 동영상의 전반부인 프리픽스를 사용자에게 전송하여 초기 지연시간을 최소화하고 이와 동시에 원거리에 위치한 주 서버로부터 후반부인 서픽스(suffix)를 전송받는다. 이와 같은 프리픽스 저장 기법을 기반으로 한 최근의 연구에서 사용자들에 의해 요청된 데이터의 전체 크기에 따라 정해진 인기도를 이용하는 PPC(Popularity-based Prefix Caching)[8] 방법이 제안되었다. 제안된 PPC 방법은 인기도에 따라서 비디오 프록시 서버에 저장될 동영상 데이터의 크기를 결정하고 해당 크기의 앞부분 동영상 데이터를 우선하여 저장한다. 그러나 이 방법은 접근 최근성(recentness)을 전혀 고려하지 않기 때문에 과거에 자주 요청되었던 동영상들로 인해 새롭게 요청되고 있는 동영상 데이터가 비디오 프록시 서버에 저장되지 못하는 문제점이 발생한다. 또 다른 연구에서 미리 설정된 횟수의 사용자 요청 시간 정보를 활용하는 LRU-k 알고리즘[9][10]과 사용자의 요청에 따른 우선순위를 고려하여 비디오 프록시 서버에 동영상의 저장과 삭제를 결정하는 Greedy-Dual 알고리즘[11]이 제안되었다. 그러나 이러한 방법들은 접근 빈도수나 접근 최근성을 주로 활용한 방법으로 동영상 데이터의 사용자 요청과 관련된 정보들을 충분히 고려하고 있지 않다. 특히 사용자가 요청한 모든 동영

상 데이터들이 비디오 프록시 서버에 저장되므로 한번 저장된 후 계속 요청되지 못하는 동영상 데이터 역시 비디오 프록시 서버에 저장되게 된다. 이는 곧 새로운 동영상 데이터의 저장을 위한 공간 부족 시에 과도한 동영상 데이터의 삭제가 수행되는 원인이 된다. 이러한 문제를 효율적으로 해결하기 위하여 본 논문에서는 사용자가 요청한 동영상 데이터를 비디오 프록시 서버의 주기억장치에 일단 적재한 후 해당 동영상이 설정된 시간 내에 지속적으로 요청되는 경우 이를 비디오 프록시 서버에 저장하는 시간 제약 다중 요청 기법을 제안한다. 제안하는 방법은 사용자가 요청한 동영상 데이터들 중 사용자에게 계속 요청될 데이터들만을 선별하여 비디오 프록시 서버의 저장 공간으로 이동하게 한다. 따라서 사용자들은 비디오 프록시 서버에 저장되어있는 동영상들을 빈번하게 요청하게 되며 비디오 프록시 서버는 해당 동영상 데이터를 사용자에게 신속하게 전송할 수 있게 됨으로써 원거리에 위치한 주 서버로부터의 데이터 전송으로 인한 과도한 네트워크 트래픽이나 서비스 지연으로 인한 낮은 전송 품질의 문제점을 해결한다. 또한 비디오 프록시 서버에 저장되어 있는 동영상을 대상으로 주로 요청되는 세그먼트 그룹인 전방 클래스(forward class)와 요청될 가능성이 상대적으로 적은 후방 클래스(backward class)로 구분하여 삭제를 수행함으로써 비디오 프록시 서버의 제한된 용량을 효율적으로 관리한다.

본 논문의 구성은 다음과 같다. II장에서는 효율적인 인터넷 스트리밍 서비스를 위한 동영상 데이터 관리 기법을 제안한다. 또한 III장에서는 제안하는 방법의 효율성을 검증하기 위해 실험을 수행하고 그 결과를 분석한다. 마지막으로 IV장에서 본 논문의 결론과 전반적인 내용에 대한 요약의 기술한다.

II. 시간제약 다중 요청 기법 기반 동영상 데이터 관리

본 논문에서는 그림 1과 같이 비디오 프록시 서버를 사용자와 근거리에서 위치시키고 이를 활용하여 주 서버의 부하 감소와 초기 전송 지연 및 동영상 패킷 손실의 문제점을 해결하고자 한다.

비디오 프록시 서버의 저장 공간인 저장장치는 원격지의 주 서버에 비해 상대적으로 작은 저장 공간을 가진다. 따라서 비디오 프록시 서버는 사용자에 의해 빈번하게 요청될 것으로 판단된 동영상만을 선별하여 저장장치에 저장하고 이를 사용

자에게 전송해야 한다. 그러나 시간에 따른 사용자의 동영상 요청 패턴이 변화함에 따라서 지속적으로 요청될 것으로 판단되어 비디오 프록시 서버에 저장되어있는 동영상이 사용자에게 의해 더 이상 요청되지 않아 삭제될 필요성이 발생하게 되며, 이 때 비디오 프록시 서버의 효율적인 저장 공간 관리가 요구된다. 이를 위해 본 논문에서는 새로운 동영상을 비디오 프록시 서버의 저장장치에 효율적으로 저장하고 기존에 저장된 동영상을 선별하여 제거함으로써 여유 공간을 확보하는 방법을 제안한다. 동영상의 효율적인 저장과 전송을 위해 본 논문에서는 하나의 동영상을 여러 개의 세그먼트로 구성하는 미디어 분할 기법을 활용한다. 세그먼트의 분할은 chen et al.[12]이 제안한 지속적으로 증가하는 방법을 고려한다. 또한 사용자의 동영상 요청 시에 동영상의 전반부인 프리픽스를 저장장치에 저장하는 프리픽스 저장 기법을 활용한다.

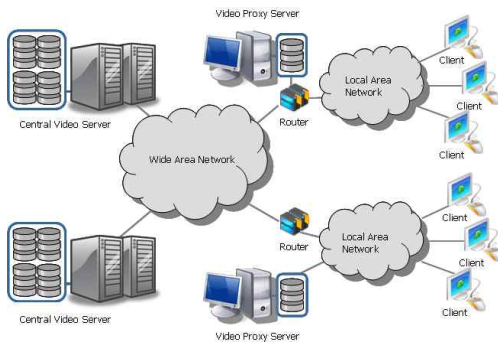


그림 1. Vod 시스템 구조
Fig 1. Architecture of VoD System

제약하는 시간 제약 다중 요청 기법은 사용자에게 의해 빈번하게 요청될 것이라고 예측된 동영상의 세그먼트만을 선별적으로 저장장치에 저장하는 방법이다. 이를 위해 비디오 프록시 서버는 사용자가 요청한 동영상의 세그먼트들을 사용자 요청이 중단될 때까지 유지시키며 이후의 사용자가 해당 세그먼트를 요청한 경우에 저장장치에 저장한다. 그림 2(a)와 같이 사용자는 관심 동영상에 대해 세션(session)을 설정하고 세그먼트들을 요청한다. 비디오 프록시 서버는 사용자가 요청한 세그먼트를 유지시키고 미리 결정된 크기의 프리픽스에 해당하는 세그먼트를 저장장치에 저장한다. 프리픽스의 크기를 결정하기 위해 주어진 시간 간격(Δ)동안 비디오 프록시 서버는 주 서버와의 동영상 전송 시 발생한 초기 지연 시간을 지속적으로 측정한다. 측정된 값들을 대상으로 평균값을 계산하고 이에 해당하는 프리픽스의 크기를 결정한다. 세션이 중단되기 전에 해당 동영상에 대한 새로운 세션이 설정되지 않는다면 저장되어 있는 모든 세그먼트는 삭제된다.

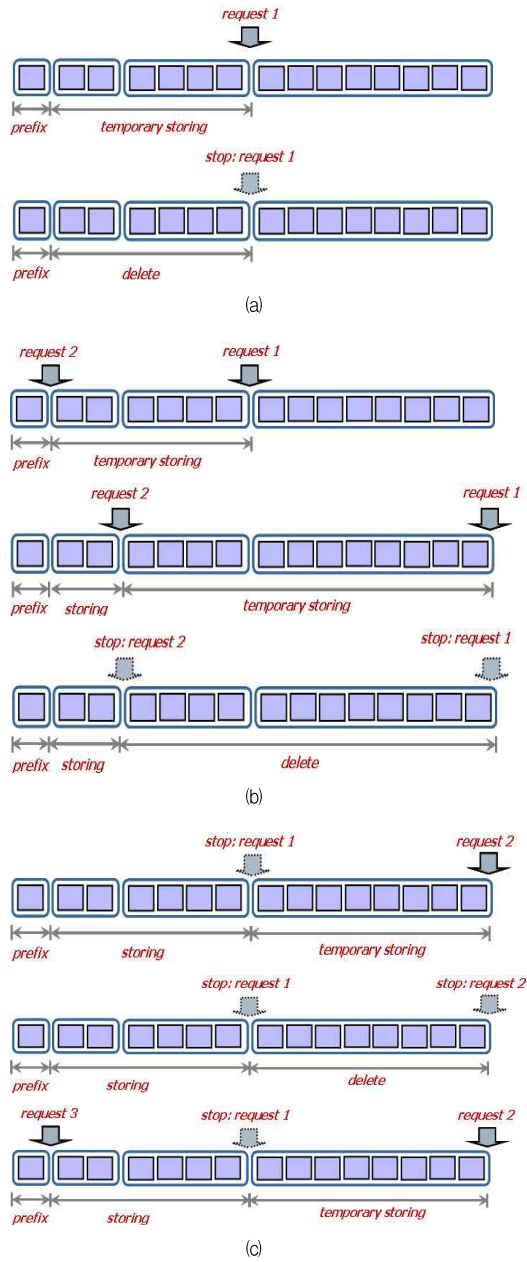


그림 2 시간 제약 다중 요청 기법 : (a) 단일세션, (b) 저장을 위한 다중세션, (c) 저장과 유지를 위한 다중 세션
Fig 2. Time constraint multiple access technique : (a) single session, (b) concurrent session for storing, (c) concurrent session for storing and maintenance

만일 새로운 세션이 설정되고 일시적으로 저장되어 있는 세그먼트를 요청하게 되면 그림 2(b)과 같이 비디오 프록시 서버는 해당 세그먼트를 저장장치에 저장한다. 그러나 사용자에게

의해 요청되지 않을 경우 해당 세그먼트는 삭제된다. 그림 2(c)에서와 같이 일시적으로 저장되어있는 동영상 세그먼트가 연결된 세션에 의해 모두 요청된 경우 비디오 프록시 서버는 해당 세그먼트들을 저장장치에 저장하고 이후에 새롭게 요청된 세그먼트는 일시적으로 저장하고 사용자의 요청을 기다린다.

제한하는 시간 제약 다중 요청 기법은 임의의 동영상에 대해 첫 번째 요청 세션이 중단되기 전에 새로운 세션이 설정되어야만 해당 동영상의 세그먼트를 저장하는 것이 가능하다. 만일 임의의 동영상에 대해 사용자의 요청이 감소하여 첫 번째 요청 세션이 중단되기 전에 새로운 세션이 설정되지 않는다면 해당 동영상은 어떠한 세그먼트도 저장되지 못한다. 따라서 본 논문에서는 저장 여유 공간에 따라 이전 시간 간격동안 각 동영상이 요청된 시간 간격들의 평균값인 평균 요청간격(mean request interval)을 활용하여 해당 간격동안 중단된 세션을 유지시키는 가상 연결(virtual bridge) 기법을 제안한다. 그림 3은 평균 요청 간격 값을 갖는 가상 연결을 활용하여 시간 제약 다중 요청 기법을 수행하는 것을 보인다.

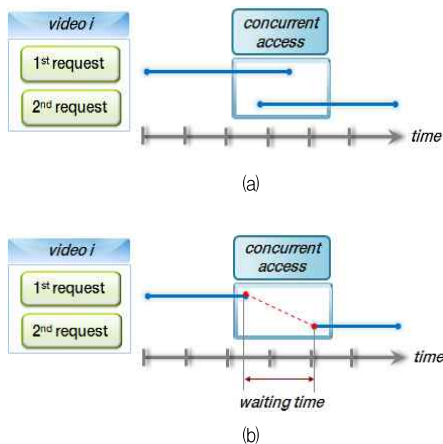


그림 3. 세그먼트 저장을 위한 대기시간: (a) 대기시간 = 0, (b) 대기 시간 = 평균 요청 간격
 Fig. 3. Waiting time for segment storing: (a) waiting time = 0, (b) waiting time = average request interval

비디오 프록시 서버는 원격지의 주 서버에 비해 상대적으로 작은 저장 공간을 가진다. 따라서 새롭게 요청된 동영상의 저장 공간을 확보하기 위해 저장장치에 저장되어 있는 동영상 세그먼트 중 요청 가능성이 가장 낮은 세그먼트를 선정하고 제거해야 한다. 만일 새로운 세그먼트의 입력으로 인해 저장 장치의 공간 부족 현상이 발생할 경우 발생한 시점에서 공간을 확보하는 것은 비효율적이다. 따라서 비디오 프록시 서버 저장장치 여유 공간이 저장되어있는 가장 큰 세그먼트(s) 보

다 작을 경우에 공간 확보 알고리즘을 수행한다. 본 논문에서는 사용자에게 의해 주로 요청되는 동영상 세그먼트 부분과 요청되지 않았거나 요청될 가능성이 적은 세그먼트로 분리하기 위해 하나의 동영상 세그먼트들을 전방 클래스와 후방 클래스로 구분한다. 여기서 전방 클래스에 속한 세그먼트들을 비디오 프록시 서버의 저장 공간에 우선 저장하고 공간 부족 현상 발생 시에 후방 클래스에 속한 세그먼트들을 대상으로 하여 우선 제거를 수행한다. 이를 위해 식 (1)을 활용하여 임의의 동영상 i 를 대상으로 가장 큰 요청 횟수 차이 값을 가지는 세그먼트를 중심 세그먼트로 정하고 해당 세그먼트를 기준으로 앞부분에 위치한 세그먼트들을 전방 클래스로 하고 뒷부분에 위치한 세그먼트들을 후방 클래스로 정의한다. 세그먼트 간의 요청 횟수 차이값이 크다는 의미는 차이가 발생하는 세그먼트 까지 주로 요청되고 이후에 사용자의 요청이 빈번하게 중지되었다는 의미이다.

$$\lambda_i(k) = \{Pop_i(k) - Pop_i(k+1)\}, k=1, 2, \dots, n \dots\dots\dots (1)$$

여기서 $\lambda_i(k)$ 는 임의의 동영상 i 의 세그먼트 k 가 가지는 요청 횟수 차이값이며 $Pop_i(k)$ 는 임의의 동영상 i 의 세그먼트 k 가 가지는 요청 횟수이다. 또한 n 은 임의의 동영상 i 가 가지는 세그먼트 개수이다.

만일 전방 클래스로 구성된 저장 장치의 동영상들의 마지막 세그먼트가 설정된 대기 시간동안 요청되지 않을 경우 후방 클래스로 이동하여 삭제 대상 후보로 한다. 식 (2)는 대기 시간의 계산을 보인다.

$$\mu_i(k) = \frac{1}{n} \sum_{i=1}^n a_i(k) \dots\dots\dots (2)$$

여기서 $\mu_i(k)$ 는 임의의 동영상 i 의 마지막 세그먼트 k 에 할당된 대기시간이며 n 은 저장 장치에 존재하는 동영상들 중 세그먼트 k 를 가지는 동영상의 개수이다. 또한 $a_i(k)$ 는 임의의 동영상 i 의 세그먼트 k 가 가장 최근에 요청된 시간과 해당 시간 이전에 요청된 시간 간의 간격이다.

비디오 프록시 서버 저장장치의 동영상들을 대상으로 가장 오래전에 요청된 동영상을 선택하고 해당 동영상의 후방 클래스의 마지막 세그먼트부터 공간 확보가 완료될 때까지 지속적으로 삭제를 수행한다. 이때 후방 클래스의 크기가 s 의 크기보다 클 경우 해당 클래스의 모든 세그먼트를 한 번에 모두 삭제하여 전체적인 삭제 횟수를 감소시킨다. 만일 선택된 동영상의 후방 클래스가 더 이상 존재하지 않을 경우 전방 클래스의 마지막 클래스부터 삭제하여 저장 공간을 확보한다. 이렇게 함으로써 사용자가 주로 요청하는 전방 클래스의 세그먼트

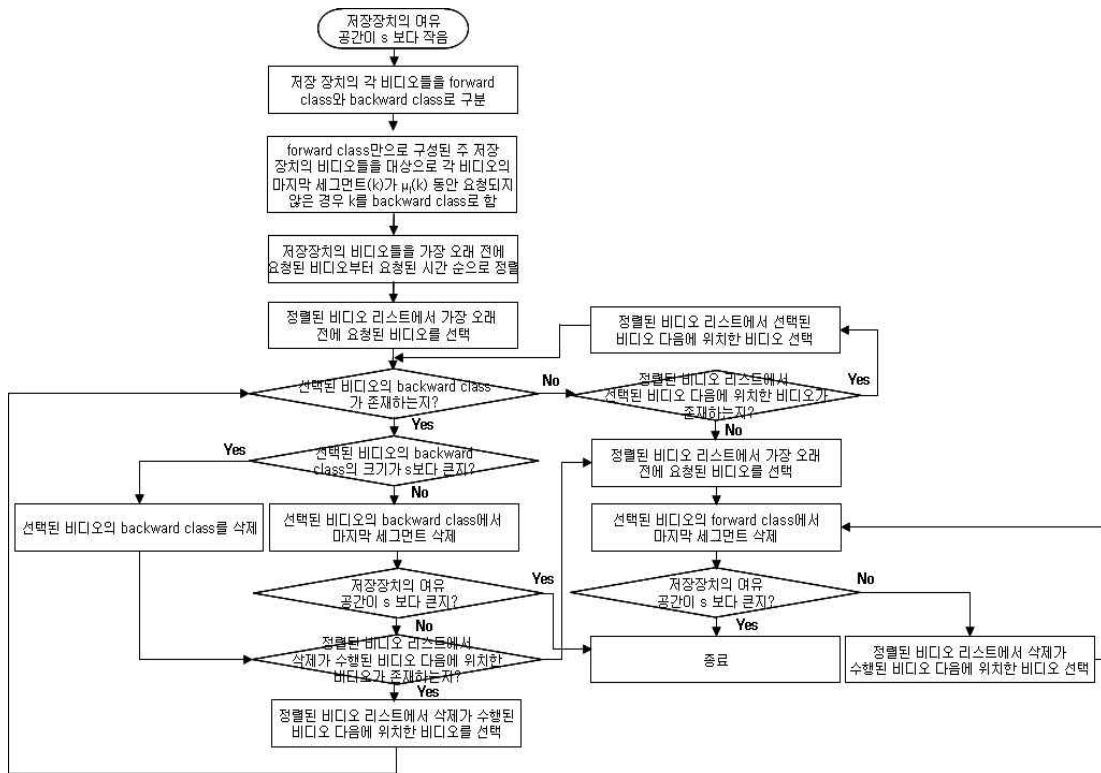


그림 4. 동영상 세그먼트 삭제 알고리즘
Fig 4. Flowchart of the segment deletion

트들은 비디오 프록시 서버에 가능한 오랜 시간동안 저장하는 동시에 사용자의 요청이 빈번하게 이루어지지 않은 후방 클래스의 세그먼트들을 우선 삭제함으로써 비디오 세그먼트의 저장 및 삭제의 효율을 높인다. 그림 4는 본 논문에서 제안하는 동영상 세그먼트 삭제 알고리즘을 보이는 플로차트이다.

III. 실험

제안하는 알고리즘의 성능 평가를 위해서 기존의 알고리즘인 PLFU(Partial Least Frequently Used)[13]와 Distance-based 방법[14] 그리고 Reallocation Affinity [15] 방법을 대상으로 비디오 프록시 서버의 저장 공간 크기 변화에 따른 블록 적중률(block hit rate)과 블록 삭제 횟수(number of block deletion)를 비교한다. 이를 위하여 본 논문에서는 그림 5와 같이 주 서버에서 동영상을 여러 개의 세그먼트(segment) 단위로 분할하여 전송하는 Chen 등이 제안한 피

라미드식 미디어 분할(media segmentation) 기법[12]이 사용된다. 미디어 분할 기법을 사용하면 적은 비용으로 많은 데이터를 신속하게 삭제하거나 저장할 수 있으므로 초기 지연과 네트워크의 트래픽을 효과적으로 줄여 주는 장점이 있다. 실험을 위해 분 당 1,200회의 요청 횟수로 사용자가 원하는 동영상 데이터가 요청된다. 표 1은 제안하는 방법의 효율성을 확인하기 위한 실험을 수행하기 위한 조건을 보인다.



그림 5. 미디어 분할 피라미드 분할
Fig 5. Media segmentation approach: pyramid segmentation

표 1. 실험조건
Table 1. Simulation parameters

항목	값
Simulation time (T)	72 hours
Number of videos	1200 files
Video size	654.7MB
Bit rate	1024Kbps
Block play time	5 sec.
Storage size of video proxy server (GB)	50, 100, 150, 200, 250, 300, 350
Segmentation method	Pyramid segmentation
Number of segment	10 segment per video

그림 6은 본 논문에서 제안하는 시간 제약 다중 요청 기법을 활용하여 기존의 방법인 PLFU와 Distance-based 방법 그리고 Reallocation Affinity 방법과 블록 적중률의 비교를 보인다. 실험 결과를 통해 기존의 방법들에 비해 비교적 높은 블록 적중률을 보임으로서 제안하는 비디오 프록시 서버에서의 시간 제약 다중 요청 기법이 사용자에 의해 향후 계속 요청 될 동영상 데이터들을 선별하여 저장하고 있음을 알 수 있다.

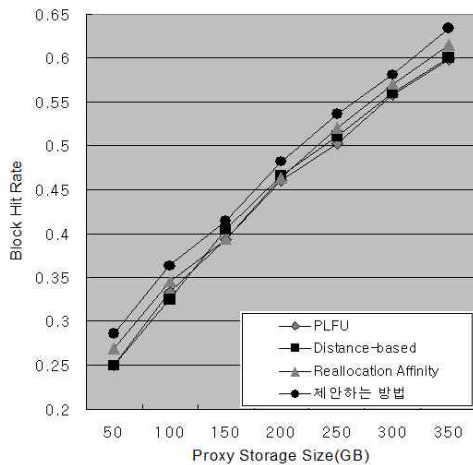


그림 6. 비디오 프록시 서버 저장 공간의 크기 변화에 따른 기존 알고리즘의 블록 적중률 비교
Fig 6. Comparison of block hit rate under various video proxy server storage size

제안하는 방법의 효율성을 보다 정확하게 확인하기 위해 그림 7과 같이 비디오 프록시 서버 저장 공간의 크기 변화에 따른 블록 삭제 횟수를 실험한다. 실험 결과를 통해 제안하는 방법이 기존 방법들에 비해 비교적 적은 블록 삭제 횟수를 보이는 것을 확인할 수 있다. 이를 통해 제안하는 방법이 사용

자에 의해 요청될 가능성이 적은 동영상 데이터들만을 대상으로 삭제 알고리즘을 수행하여 자주 요청하는 동영상의 세그먼트들만을 비디오 프록시 서버에 유지하도록 함으로서 비디오 프록시 서버의 저장 공간에서 수행되는 블록 삭제 횟수를 효과적으로 감소시키고 있음을 알 수 있다.

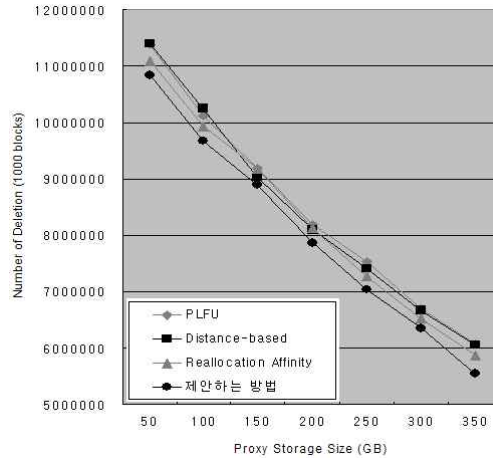


그림 7. 저장장치의 크기 변화에 따른 기존 알고리즘과의 교환 횟수 비교

Fig 7. Comparison of number of replacement under various storage size

IV. 결론

현재의 동영상 전송은 손실과 지연에 상당히 민감하게 반응하여 이를 기반으로 한 여러 분야의 발전에 커다란 문제점으로 지적되고 있다. 이와 같은 문제를 해결하기 위해 사용자들과 근거리에 위치한 비디오 프록시 서버를 활용하여 인터넷 VoD(Video-on-Demand) 서비스의 품질을 크게 개선시킬 수 있다. 그러나 비디오 프록시 서버는 비교적 제한된 용량을 가진다는 단점이 있다. 따라서 본 논문에서는 원격지에 위치하는 주 서버에 비해 상대적으로 제한된 저장 공간을 가지는 비디오 프록시 서버에 사용자들이 계속적으로 요청하게 되는 동영상 데이터만을 저장하는 시간 제약 다중 요청 기법을 제안한다. 제안하는 기법은 사용자가 요청한 동영상 데이터를 주 서버로부터 전송받아 이후 사용자들이 요청한 시각의 최근성에 기초하여 해당 동영상 데이터를 비디오 프록시 서버에 이동하거나 삭제한다. 비디오 프록시 서버의 저장 공간을 변화시켜가며 수행한 실험을 통해 제안하는 방법은 기존의 방법들에 비해서 높은 블록 적중률과 적은 블록 삭제 횟수를 나타냄으로써 보다 더 효율적임을 확인한다.

참고문헌

- [1] NimKar. A, mandal. C, and Reade C, "Video placement and disk load balancing algorithm for VoD proxy server," Proc. of IEEE Int. Conf. Internet Multimedia Services Architecture and Applications, pp. 1-6, Dec. 2009.
- [2] Jun Wu and Ravindran K, "Optimization algorithms for proxy server placement in content distribution networks," IEEE Int. Symp. on Integrated Network Management Workshop, pp. 193-198, June 2009.
- [3] BeomGu Kang, EunJo Lee, SungKwon Park, and HoSook Lee, "Popularity-based partial caching management scheme for streaming multimedia on proxy servers over IP networks," Proc. of IEEE Int. Conf. Network Infrastructure and Digital Content, pp. 586-590, Nov. 2009.
- [4] Chakareski. J, "Efficient proxy-driven multiple user video streaming," IEEE Int. Symp. on Multimedia Signal Processing," pp. 1-6, Oct. 2009.
- [5] 김은삼, "대규모 멀티미디어 서버에서 효율적인 데이터 이동/중복 기법," 한국컴퓨터정보학회논문지, 제 14권, 제 5호, 37-44쪽, 2009년 5월.
- [6] 정현식, "VOD 시스템에서의 실시간 응용 프로그램을 위한 디스크 배열 구조의 성능 분석," 한국컴퓨터정보학회 논문지, 제 5권, 제 4호, 106-111쪽, 2000년 4월.
- [7] Kaihui Li, Changqiao Xu, Yuanhai Zhang, and Zhimei Wu, "Optimal Prefix Caching and Data Sharing Strategy," Proc. of IEEE Int.Conf. Multimedia and Expo, pp. 465-468, June 2008.
- [8] Seong Ho Park, Eun Ji Lim, Ki Dong Chung, "Popularity-based partial caching for VOD systems using a proxy server," Proc. of Parallel and Distributed Processing Symposium, pp. 1164-1168, April 2001.
- [9] E. J. O'Neil, P.E. O'Neil, G. Weikum. "The LRU-K Page Replacement Algorithm For Database Disk Buffering," Proc. of the 1993 ACM SIGMOD Conference, pp. 297-306, 1993.
- [10] S. Acharya and B. Smith, "MiddleMan: A Video Caching Proxy Server," Proc. of 10th International Workshop on Network and Operating Systems Support for Digital Audio and Video, Chapel Hill, NC, June 2000.
- [11] S. Jin and A. Bestavros, "Popularity-aware greedy dual-size web proxy caching algorithms," Proc. of 20th IEEE Intl. Conf. Distributed Computing Systems (ICDCS), Taipei, Taiwan, pp. 254-261, April 2000.
- [12] Songqing Chen, Haining Wang, Xiaodong Zhang Shen B, and Wee S, "Segment-based proxy caching for Internet streaming media delivery," IEEE Trans. on Multimedia, Vol. 12, pp. 59-67, Sept. 2005.
- [13] Kuan-Sheng Hsueh and Sheng-De Wang, "A Packet-Based Caching Proxy with Loss Recovery for Video Streaming," Proc. of 2002 Pacific Rim Int. Symposium on Dependable Computing, pp. 185-190, Dec. 2002.
- [14] Songqing Chen, Bo Shen, Wee. S, and Xiaodong Zhang, "Segment-based streaming media proxy: modeling and optimization," IEEE Trans. on Multimedia, Vol. 8, pp. 243-256, April 2006.
- [15] Christian Spielvogel and Laszlo Boszormenyi, "Quality-of-Service based Video Replication," Proc. of Int. Workshop on Semantic Media Adaptation and Personalization, pp. 21-26, Dec. 2007.

저 자 소 개



이 준 표
2008년: 한양대학교 공학박사
2001~2003년: 파인드테크(주)
연구원
2003~2009년: 한양대학교 공학
기술연구소 연구원
2009~ 현재: LIG넥스원(주) 선임
연구원
관심분야: 영상처리, 멀티미디어
시스템, 패턴인식



조 철 영
2008년: 충남대학교 공학사.
2008~ 현재: LIG넥스원(주) 연구원
관심분야: 패턴인식, 지능제어, 인지과학



권 철 희
2000년: 고려대학교 공학석사
2000~ 현재: LIG넥스원(주)
책임연구원
관심분야: 패턴인식, HCI, 디지털 신호
처리, 컴퓨터 기술 응용



이 중 순
1982년: 한양대학교 이학박사
1984년: 한양대학교 이학석사
1986~ 현재: LIG넥스원(주)
수석연구원(팀장)
관심분야: 시스템 제어, 정보 및 영상
신호처리



김 태 영
1985: 경북대학교 공학사
1987: 경북대학교 공학석사
2009: 성균관대학교 공학박사
1990~ 현재: LIG넥스원(주)
수석연구원(센터장)
관심분야: 디지털신호처리, 컴퓨터응용