

클러스터 VOD 서버에서 선호도 기반 세그먼트 버퍼 대체 기법

(Preference-Based Segment Buffer Replacement in Cluster VOD Servers)

서 동 만 [†] 이 좌 형 [†] 방 철 석 ^{**}
(Dongmahn Seo) (Joahyoung Lee) (Cheolseok Bang)

임 동 선 [†] 정 인 범 ^{***} 김 윤 ^{***}
(Dongsun Lim) (Inbum Jung) (Yoon Kim)

요 약 대규모 사용자들에게 QoS가 보장되는 스트리밍 미디어를 제공하기 위해서는 한정된 서버의 자원을 스트리밍 미디어의 특징에 맞추어 효과적으로 활용하는 연구를 필요하게 한다. 서버의 자원들 중 메모리는 디스크로부터 읽어 들인 미디어 데이터의 버퍼공간으로 활용되며 버퍼의 히트율은 서버의 성능에 중요한 역할을 한다. 그러나 접속되는 사용자들의 증가에 따라서 히트율이 높은 기존의 버퍼들조차도 새롭게 읽어 들인 미디어 데이터들로 대체되므로 서버의 성능확장성에 부정적 영향을 미치는 결과를 가져오고 있다. 이런 문제점을 해결하기 위해서는 스트리밍 미디어의 특징인 대규모 데이터들에 대한 읽기 연속성과 편중된 사용자 선호도가 반영된 버퍼 대체 알고리즘이 필요하다. 본 논문에서는 대규모 사용자들에게 스트리밍 미디어를 서비스하기 위하여 사용되는 클러스터 기반 VOD 서버에서 스트리밍 미디어의 특징을 활용하는 선호도 기반 세그먼트 버퍼 대체 알고리즘을 연구한다. 제안되는 기법은 사용자들의 선호도에 기반을 둔 시간적 국부성과 미디어 데이터의 순차적 읽기 특성인 공간적 국부성을 버퍼 대체 알고리즘에 반영하므로 기존의 버퍼 대체 알고리즘보다 스트리밍 미디어 서비스에서 향상된 버퍼 히트율을 나타낸다. 본 논문에서는 향상된 버퍼 히트율이 클러스터 기반 VOD 서버의 성능 확장성을 개선시킴을 구현된 클러스터형 VOD 시스템을 통하여 입증한다.

키워드 : 선호도, 버퍼 대체, 히트율, 세그먼트, 스트리밍 미디어, 확장성

Abstract To support the QoS streams for large scale clients, the internal resources of VOD servers should be utilized based on the characteristics of the streaming media service. Among the various resources in the server, the main memory is used for the buffer space to the media data loaded from the disks and the buffer hit ratio has a great impact upon the server performance. However, if the buffer data with high hit ratio are replaced for the new media data as a result of the number of clients and the required movie titles are increased, the negative impact on the scalability of server performance is occurred. To address this problem, the buffer replacement policy considers the intrinsic characteristics of the streaming media such as the sequential access to large volume data and the highly disproportionate preference to specific movies. In this paper, the preference-based segment buffer replacement policy is proposed in the cluster-based VOD server to exploit the characteristics of the streaming media. Since the proposed method reflects both the temporal locality by the clients' preference and the spatial locality by the sequential access to media data, the buffer hit ratio would

· 본 연구는 한국과학재단 목적기초연구(R05-2003-000-12146-0)의 지원으로 수행되었음

· 2004년도 강원대학교 학술연구조성비로 연구하였음
· 이 논문은 강원대학교 정보통신연구소에 의해 지원되었음

[†] 학생회원 : 강원대학교 컴퓨터정보통신공학과
dmseo@snslab.kangwon.ac.kr
jhlee@snslab.kangwon.ac.kr
dslim@snslab.kangwon.ac.kr

^{**} 정 회원 : TID 연구팀 연구원

csbang@snslab.kangwon.ac.kr

^{***} 종신회원 : 강원대학교 컴퓨터정보통신공학과 교수
ibjung@snslab.kangwon.ac.kr
(Corresponding author)

^{****} 정 회원 : 강원대학교 컴퓨터정보통신공학과 교수
yooni@kangwon.ac.kr

논문접수 : 2006년 3월 22일

심사완료 : 2006년 8월 21일

be improved as compared to the existing buffer replacement policy. The enhanced buffer hit ratio causes the fact that the performance scalability of the cluster-based VOD server is linearly improved as the number of cluster nodes is increased.

Key words : preference, segment, buffer replacement, hit ratio, streaming media, scalability

1. 서론

최근 컴퓨터 기술과 정보통신망의 발전에 따라 멀티미디어 데이터에 대한 요구가 증가함은 물론 다양해지고 있다. 멀티미디어 데이터 들 중 스트리밍 미디어는 사용자들이 장소 및 단말에 제한 받지 않고 서버로부터 전송되는 동영상 서비스를 즐길 수 있다는 점에서 가장 각광받는 멀티미디어 서비스로 자리매김하고 있다. 스트리밍 미디어 서비스에서는 서버에 저장되어있는 수많은 동영상 미디어들 중 사용자는 자신이 원하는 미디어를 선택하여 즉시 시청할 수 있을 뿐만 아니라 전송되는 미디어는 사용자에게 QoS를 제공해야 한다[1-3].

스트리밍 미디어 서비스를 사용하는 동시 접속자수가 늘어남에 따라 모든 사용자에게 좋은 품질의 서비스를 하기 위해 고성능, 고용량의 서버가 필요로 하게 된다. 그러나 아무리 대규모의 서버 시스템일지라도 서버의 자원을 무한정 확장하는 것은 불가능하기에, 서버의 한정된 내부 자원을 효과적으로 이용하기 위한 연구가 필요하다. 서버의 자원을 크게 메모리와 네트워크, CPU, 디스크로 나누어 볼 때, 각각의 자원을 효과적으로 사용하는 것은 전체적인 스트리밍 미디어 서비스의 성능에 큰 영향을 미친다. 이러한 서버 자원들 중 서버의 성능 확장성에 제약이 되는 자원으로는 메모리, 네트워크의 대역폭, 디스크이다. 특히 고가이면서 적은 용량을 가지는 메모리는 대용량의 미디어 데이터를 다루는 스트리밍 미디어 서버 성능에 중요한 역할을 하고 있다[1,4]. 제한된 메모리를 효과적으로 이용하기 위해 서버 내 버퍼의 히트율을 높여야한다. 즉 디스크에서 읽어온 미디어 데이터는 버퍼에 저장되므로 버퍼에 저장된 미디어 데이터를 다음 사용자들도 사용하게 함으로 버퍼의 히트율을 높이고 결과적으로 스트리밍 미디어 서버의 성능 향상에 중요한 요인이 되고 있다.

대규모의 사용자들에게 스트리밍 미디어 서비스가 진행되는 경우 특정한 시간대에 인기 있는 미디어들에 대한 요청이 집중적으로 발생하는 특징을 보이고 있다. 또한, 스트리밍 미디어는 영화의 진행 특성상 콘텐츠에 대한 읽기가 영화 재생 순서에 따라 연속적으로 발생하는 특징을 가지고 있다. 이러한 환경에서 디스크로부터 읽어 들인 미디어 데이터는 버퍼공간에 순차적으로 적재되게 된다. 많은 사용자들로부터 선호도가 높은 영화 미디어의 경우 버퍼에 적재된 상태로 히트 되므로 디스크

로부터 읽어 들임 없이 바로 스트리밍 서비스를 하므로 미디어 서버의 성능에 절대적인 역할을 하게 된다. 그러나 접속되는 사용자들의 증가와 요구하는 영화 콘텐츠의 증가는 히트율이 높은 기존의 버퍼들이 새롭게 읽어 들인 미디어 데이터들로 대체될 경우 곧이어 버퍼 미스가 발생하므로 서버의 성능확장에 부정적 영향 가져오게 된다. 이런 문제점을 해결하기위해서는 스트리밍 미디어의 특징인 대규모 데이터들에 대한 순차적 읽기 연속성 및 편중된 사용자 선호도가 반영된 버퍼 대체 알고리즘이 필요하다.

본 논문에서는 스트리밍 미디어 서비스를 하는 클러스터 서버에서 효과적인 버퍼 관리를 위하여 사용자의 편중된 선호도와 영화 미디어 데이터의 순차적 읽기 특성을 고려한 버퍼대체 알고리즘을 제안하고자한다. 클러스터 서버는 가격대 성능비가 뛰어난 서버 시스템으로 웹서버, VOD 서버 및 게임 서버 등에서 사용되고 있다. 스트리밍 미디어로는 MPEG 규격의 미디어들이 사용되며 클러스터 서버의 성능 확장성을 위하여 선택된 MPEG 미디어들은 병렬 처리의 성질도 단위로 분할되어 클러스터 노드들에 저장된다. VOD 시스템에 접속된 사용자가 영화 미디어를 요구하면 분산되어 저장된 미디어 데이터들은 각각의 클러스터 노드들로부터 동시에 병렬로 인출되어 스트리밍 서비스를 시작한다. 다수의 사용자들이 동일한 영화의 데이터를 요구하는 경우 디스크가 아닌 버퍼에 기 적재된 해당 미디어의 데이터들로 스트리밍 서비스를 우선적으로 진행할 수 있으므로 버퍼 히트율을 높이는 방법은 서버 전체의 성능 향상에 큰 역할을 한다. 본 논문에서 제안되는 버퍼대체 기법은 MPEG 스트리밍 미디어의 특징을 활용하는 선호도 기반 세그먼트 버퍼 대체 알고리즘이다. 제안되는 기법은 사용자들의 편중된 선호도에 기반을 둔 시간적 국부성과 미디어 데이터의 순차적 읽기 특성인 공간적 국부성을 버퍼 대체 알고리즘에 반영하므로 기존의 버퍼 대체 알고리즘보다 스트리밍 미디어 서비스에서 향상된 버퍼 히트율을 나타낸다. 구현된 클러스터 VOD 시스템을 통하여 제안된 선호도 기반 세그먼트 버퍼 대체 정책이 버퍼 히트율을 향상시킴을 관찰하며, 이러한 버퍼 히트율이 클러스터 기반 VOD 서버에서 노드의 증가에 따라서 성능 확장성을 선형적으로 개선함을 입증한다.

본 논문의 구성은 다음과 같다. 2장에서는 클러스터

서버에서의 스트리밍 미디어 서비스를 위한 기존연구 및 기존의 버퍼 대체 알고리즘에 대하여 알아본다. 3장에서는 본 논문에서 제안하는 선호도 기반 세그먼트 버퍼 대체 정책에 대해 설명한다. 4장에서는 실험환경 및 성능 측정 방법에 대하여 기술한다. 5장에서는 구성된 클러스터 VOD 서버에서 본 논문에서 제안된 버퍼 정책 및 기존의 버퍼 대체 정책에 대한 성능평가를 진행한다. 마지막으로 6장에서 결론을 맺는다.

2. 관련 연구

2.1 MPEG 미디어의 병렬처리

클러스터형 VOD 서버에서 각 노드 또는 노드 내의 각 디스크에 영화를 분산 저장하는 것을 스트라이핑이라고 한다. 스트라이핑은 각 노드간의 부하 분산과 영화 인기도에 밀접한 관련이 있다. 한 영화 데이터를 스트라이핑 하는데 있어서 특정 노드에 많은 데이터를 저장한다면, 이 영화가 요청될 경우 노드들 사이의 부하에 균형을 맞출 수 없다. 따라서 영화를 분산 저장하는 스트라이핑 단계에서부터 영화를 적절히 배분하여 저장함으로써 별도의 부하 분산 없이 각 노드 간에 균등한 부하를 줄 수 있는 방법을 사용해야 한다[5-8].

MPEG 미디어는 여러 계층으로 이루어져 있지만 독립적인 재생단위인 GOP(Group of Pictures)들의 집합으로 구성되어 있다. 하나의 GOP 내에는 I, B, P의 프레임들로 이루어져있고 B, P 프레임들은 I 프레임 및 P 프레임의 내용에 종속된다. 그러나 GOP 들 사이에는 독립성이 보장되므로 GOP들을 클러스터 서버에서 MPEG 미디어를 병렬저장, 인출의 단위로 사용하여 각각의 클러스터 노드에게 스트라이핑 한다. 즉, 스트라이핑의 단위를 하나의 GOP 단위로 하여 각 노드는 순차적으로 하나의 GOP에 GOP 번호와 크기를 헤더로 첨부하여 저장한다. 따라서 영화 데이터가 초당 30프레임이고, 하나의 GOP에 15프레임이 포함되어 있다면, 각 노드는 0.5초의 상영 시간을 갖도록 스트라이핑 된다.

2.2 선호도

다수의 사용자가 다수의 콘텐츠를 스트리밍 서비스 받고자 할 때, 사용자의 선호도가 존재하게 된다. 웹 서비스에서 서버내의 각 페이지들에도 선호도가 존재하며, 특정 페이지에 대해 선호도가 편중되는 경향이 뚜렷이 나타난다. 미디어의 스트리밍 서비스에서는 웹 서비스와 유사하게 미디어에 따른 선호도를 가진다. 뉴스의 경우 짧은 시간에 집중되는 선호도 분포를 가지는데, 이는 상당수의 이용자가 최근의 소식에 관심을 가지기 때문에 최근의 뉴스에 대한 선호도가 매우 높다. 그러나 웹 서비스와는 달리 미디어 스트리밍 서비스에서는 사용하는 데이터가 대용량임과 동시에 사용자가 요구하는 미디어

의 스트리밍 품질을 일정시간 유지시켜야 하는 QoS 규격을 지켜주어야 한다. 제한된 자원을 가진 서버에서 대규모의 사용자들에게 QoS를 만족시키는 스트리밍 서비스를 하기 위해서는 사용자들의 콘텐츠 선호도 분포를 자원관리에 이용하므로 서버의 성능 확장성 향상에 기여할 수 있다. 선호도에 대한 기존 연구 조사로는 오프라인에서 비디오 대여점의 대여현황을 분석하여 결과를 제시한 바 있다. 실제 대여 현황을 이용하여 선호도를 분석해본 결과는 zipf 분포에 근접하는 것으로 나타났고, 본 논문에서도 zipf 분포를 이용하여 실험한다[9].

2.3 버퍼 대체 정책

미디어 스트리밍 서버 내의 메모리 공간은 디스크 저장장치에 비하여 현저히 부족하기 때문에 성능 향상의 제한점이 된다. 서비스 요청의 수가 늘어나면 서비스는 더 많은 버퍼 공간을 요구하지만 시스템의 메모리 공간은 물리적인 한계를 극복하기 어렵다. 이러한 물리적 한계를 극복하고, 효과적인 메모리 사용을 위한 것이 버퍼 대체 정책이다. 효과적인 버퍼 대체 정책은 버퍼의 히트율을 높여 메모리 내에 공유되는 데이터의 양을 극대화함으로써 성능을 향상시킨다.

버퍼 대체 정책은 정적 버퍼 대체와 동적 버퍼 대체로 나눌 수 있다[10-14]. 정적 버퍼 대체는 시스템의 현재 상태를 실시간으로 반영할 수 없을 뿐만 아니라 메모리 단편화 및 버퍼관리의 부하가 발생하므로 특정 목적을 제외하고는 많이 사용되고 있지 않다. 반면에 동적 버퍼 대체는 일정한 크기로 버퍼 영역을 나누어 데이터를 동적으로 할당하는 방식이다. 동적 버퍼 대체의 대표적인 방법으로 LRU(Least Recently Used)가 있다. LRU는 과거 참조거리를 기준으로 우선순위를 부여하여 대체 대상을 선정한다. 과거 참조거리가 짧을수록 높은 우선순위가 부여되므로 대체 선정에서 제외되어 버퍼 공간에 오래 잔류하고 재사용될 가능성도 높아지게 된다. LRU는 거의 모든 응용 프로그램에서 공통적으로 나타나는 시간 국부성을 이용하는 정책이므로 대부분의 환경에서 어느 정도 만족스러운 성능을 보이며 상당수의 시스템에서 버퍼 대체 정책으로 채택되어 사용되고 있다. 그러나 LRU 정책에는 가장 최근에 접근된 것이 항상 최우선순위를 가질 뿐 다른 사항은 고려하지 않기 때문에 각각의 어플리케이션 특징을 반영하지 못하는 단점이 있다. 이러한 단점을 보완하기 위하여 LRU에 대한 많은 연구가 진행되어 왔으며, 많은 변형된 LRU 정책들이 있다[1,15,16].

LRU 정책은 접근빈도에 대한 배려가 없기 때문에, 자주 사용되는 데이터와 그렇지 않은 데이터에 대한 구분이 없어 자주 사용되지 않는 데이터도 버퍼 공간에 오래 잔류하는 문제가 발생한다. 얼마 전까지 빈번한 접

근이 이루어진 것보다 최근 한 번의 접근이 이루어진 것이 보다 높은 우선순위를 가지게 된다. 이러한 단점을 해결하기 위해 LRU-K[15], LRFU[16] 등이 연구되어 기본 LRU의 단점을 어느 정도 해결하고 있다. 이 방법들의 주요 해결책은 버퍼에 카운터를 두거나 과거 참조 거리를 기반으로 대체될 버퍼를 정하는 것으로, LRU 대체 정책에 접근 빈도에 따른 변수를 추가하여 데이터에 대한 접근 빈도를 고려하는 것이다. 또한, LRU 정책에서는 과거 참조거리가 전체 버퍼의 크기를 넘을 경우 연속적인 버퍼 미스가 발생하게 된다. 버퍼의 크기가 주기적으로 접근하는 데이터의 크기보다 작기 때문에 반복적인 접근임에도 불구하고 연속적인 미스가 발생하게 된다. 이런 문제를 해결하고자 연구된 것이 EELRU 정책이다[17].

2.4 멀티미디어 데이터를 위한 동적 버퍼 관리 및 캐싱 기법

멀티미디어 데이터의 대용량성과 순차적이고 반복적인 접근 특성을 잘 반영한 것이 그룹 단위의 버퍼 관리이다. 그룹 단위 버퍼 관리란 크기가 작은 여러 개의 버퍼를 하나로 묶고 이를 하나의 기본 단위로 하여 버퍼를 관리하는 것이다[18].

하나의 그룹이 관리의 기본 단위가 되므로 어떤 조건을 가지고 그룹을 형성하는지도 중요하다. 일반적으로 시간적, 공간적으로 근접한 거리의 서비스들을 하나의 그룹으로 형성한다. 가장 단순하게 그룹을 나누는 방법은 고정적인 크기를 주기적으로 검사하는 방법이다. 특정 시간에 전체 버퍼에 대한 접근 현황을 통해 그룹을 만들고, 메모리 부족으로 대체가 필요한 경우 해당 그룹을 대상으로 대체 작업을 수행한다. 이렇게 형성된 그룹은 동적으로 관리되어 상황에 따라 서비스 종료까지 유지되거나 분할 또는 병합하여 효율성을 향상시키게 된다[19].

멀티미디어 데이터의 특성을 반영하고 서버의 성능을 향상 시키기위해 프록시 서버에서 버퍼를 사용하는 다양한 프록시 버퍼가 연구되었다[20-23]. 런닝 버퍼 캐싱(Running Buffer Caching)[20]은 고정된 크기의 버퍼를 두고 앞선 사용자의 데이터를 캐싱하여 뒤이은 사용자에게 서버를 거치지 않고 사용하도록 한다. 인터벌 캐싱(Interval Caching)[21,22] 기법은 사용자 요청의 시간차를 고려하여 버퍼 메모리를 효과적으로 사용하기 위한 방법이다. 앞선 두 캐싱 방법은 멀티미디어 데이터의 특성을 잘 반영하고 있으나, 사용자의 접근 패턴을 고려하고 있지 않다. SRB(Shared Running Buffers)[23]는 상기한 두 가지 방식의 장점과 사용자의 접근 패턴을 고려하는 방식이다. 이 방법은 사용자의 접근 패턴 중 사용자의 선호도를 이용하여 버퍼를 대체한다. 그러

나 이 방법은 실제 데이터의 특징을 고려하여 버퍼의 크기 최적화와 대용량 클러스터 VOD 서버로의 적용에 대한 문제점과 서버의 성능 확장성에 대한 고려가 부족하다.

3. 선호도 기반 세그먼트 버퍼 대체 정책

3.1 선호도 기반 버퍼 관리

스트리밍 미디어 서비스에서는 다수의 사용자들이 같은 미디어에 대한 요구들이 동시에 또는 산발적으로 발생하며, 대용량의 데이터에 대한 순차적이고 반복적인 접근 형태를 나타내게 된다. 기존의 버퍼 관리 정책들은 이러한 미디어 데이터 접근 특성을 충분히 반영하지 못해 좋은 성능을 보장하지 못한다. 이러한 대용량의 데이터에 순차적인 접근이 많은 시스템에서는 성능 개선을 위하여 지역 국부성을 고려하기 보다는 데이터에 대한 선호도 기법이나 선인출 기법 등이 더 효과적일 수 있다[19,24].

선호도를 추가한 버퍼 대체 정책은 기존의 LFU와 유사하지만 미디어 데이터의 선호도 수치에 대한 가중치를 두어서 선호도가 높은 데이터가 대체 되는 것을 한번 더 보호할 수 있게 하는 것이다. 사용 빈도에 선호도 수치를 더해주어 최근 사용빈도가 낮더라도 선호도가 높은 것은 대체 선정에서 제외시키는 것이다. 각각의 버퍼에 대한 선호도 수치를 지정하기위하여 다음 식을 사용하였다.

$$\begin{aligned} \text{favor}(i) &= \text{zipf}(i) \times N \\ \text{zipf}(i) &: I \text{ 번째 영화에 대한 요청 확률} \\ N &: \text{전체 서비스 요청 수} \end{aligned} \quad (1)$$

식 (1)에서도 알 수 있듯이 선호도 수치는 zipf 분포와 전체 서비스 요청수인 N을 곱하여 이용하였다[9]. zipf 분포는 해당 인덱스에 대한 요청 확률을 나타내기 때문에 1 미만의 실수로 나타난다. 이런 실수 값을 그대로 사용할 수 없기 때문에 정수화 과정이 필요하다.

한 번의 참조가 이루어지면 참조에 대한 가중치는 1 씩 증가하는데 선호도에 따른 차이가 1 미만이라면 선호도 수치는 의미가 없어진다. 또 실수 값을 그대로 이용한다면 계속되는 실수 연산으로 인한 성능 저하를 초래할 수도 있다. 이런 이유로 사용자의 요청에 유연하게 대응하면서 선호도 수치를 정수화하기 위해 확률 값에 N을 곱하여 가중치 수치로 사용하였다. 이 수치는 해당 스트리밍 미디어에 대해서 서비스를 요청하는 수이기도 하다. 이렇게 수치를 부여하면 전체 사용자의 요청이 증가함에 따라 자연스럽게 선호도 수치도 증가하고 반대의 경우에는 수치가 감소하여 전체 요청에 맞게 선호도 수치를 반영 할 수 있다.

3.2 세그먼트 버퍼 관리

스트리밍 미디어 서비스에서 효과적인 메모리 관리를 위해 선호도와 더불어 고려되어 져야 하는 것이 데이터에 대한 접근 특성이다. 미디어 데이터는 용량이 크고 순차적, 반복적인 접근이 이루어지기 때문에 작은 단위로 관리하는 것보다는 그 관리 기준을 크게 하는 것이 더욱 효과적이다.

스트리밍 미디어로 많이 사용되는 MPEG 데이터에서 자체적으로 재생이 가능한 기본 단위는 GOP(Group of Picture)이다. 미디어 데이터의 접근이 대용량으로 이루어진다는 점에서 볼 때 독립 재생이 가능한 기본 데이터 단위인 GOP를 여러 개 묶어서 하나의 그룹으로 관리하는 것이 더 효과적이다. 본 연구에서는 여러 개의 GOP를 묶은 하나의 세그먼트 그룹을 버퍼의 단위로 하고, 3.1장에서 설명된 사용자의 선호도를 세그먼트 별로 반영한 버퍼 대체 정책을 제안한다.

3.2.1 세그먼트 정의

세그먼트 그룹은 동일한 미디어 데이터에 대해 비슷한 부분을 서비스 받는 여러 클라이언트를 기준으로 하나 이상의 클라이언트 서비스 요청에 의해 동적으로 생성된다. 하나의 세그먼트는 1개 이상의 GOP 정보와 1개 이상의 클라이언트 서비스 요청을 가지고 있다. 세그먼트 단위로 현재 서비스가 이루어지고 있는 클라이언트 수와 버퍼의 히트 수 그리고 가능한 버퍼 수 등의 정보를 저장하고 이를 기반으로 버퍼의 대체를 수행한다.

그림 1은 세그먼트와 그 생성 과정을 보여주고 있다. 임의의 시간 t에 사용자 1의 요청이 도착한다. 이전에 만들어진 세그먼트가 있으면 자신이 포함될 수 있는지 검사한다. 자신이 처음 서비스거나 이전 세그먼트와 기준치 이상 떨어져 있는 경우 독립된 하나의 세그먼트를 생성한다. 이후 시간이 지남에 따라 사용자 2, 3, 4의 요청이 도착한다. 이렇게 하나의 미디어 데이터에 대해

여러 사용자의 요청이 있을 때 시간적으로 비슷한 시간대에 도착한 1, 2, 3은 하나의 세그먼트를 형성하게 된다. 반면에 사용자 4의 요구와 같이 기존에 형성된 세그먼트에 대하여 시간적으로 거리가 먼 서비스 요청은 독립된 다른 세그먼트를 만들게 된다. 세그먼트는 버퍼 대체 시 기준이 되어 하나의 세그먼트가 대체 될 때 해당 세그먼트 안에 있는 모든 데이터가 같이 대체된다.

3.2.2 세그먼트의 크기

세그먼트의 크기는 자원의 효율적 사용과 순차적 읽기를 진행하는 스트리밍 미디어의 특성을 고려하여 결정되어야 한다. 세그먼트의 크기를 크게 하면 하나의 세그먼트에 속하는 서비스 클라이언트 수가 증가 하게 된다. 하나의 세그먼트로 더 많은 수의 클라이언트를 대표할 수 있어 더 효과적인 것처럼 보일 수도 있지만, 세그먼트의 크기가 커지면 중간에 사용하지 않는 버퍼공간도 하나의 세그먼트 단위로 처리되기 때문에 버퍼를 세그먼트로 묶어 관리하는 효과가 반감 될 수 있다.

그림 2는 세그먼트 크기가 클 경우 발생할 수 있는 문제에 대해 보여주는 것이다. 20개의 GOP를 하나의 세그먼트로 관리하면 그림 2에 보이는 것처럼 사용자의 요구는 모두 하나의 세그먼트로 관리된다. 반면 같은 경우에 10개의 GOP를 하나의 세그먼트로 관리하면 두 개의 세그먼트로 나뉘어 관리된다. 하나의 세그먼트로 관리하게 될 경우 그림에서처럼 중간에 사용하지 않는 부분이 같은 세그먼트로 관리되기 때문에 비슷한 시간대의 접근에 대한 특성을 제대로 반영하지 못한다. 즉, 시간 국부성에 대한 특성을 살리지 못하는 것이다. 그러나 세그먼트의 크기를 너무 작게 하면 세그먼트로 묶어 공간적 국부성을 이용하려는 시도가 손상되므로 세그먼트의 크기를 정하는 것도 성능 변화에 중요한 요인이 된다.

3.3 PSBR 정책

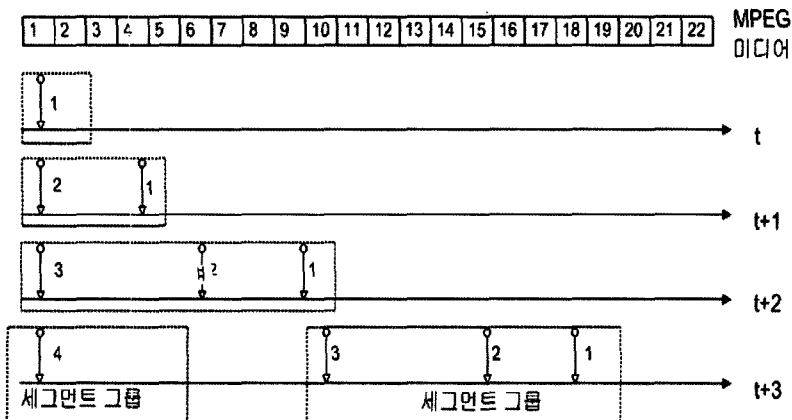


그림 1 사용자 요구에 따른 세그먼트 생성과정.

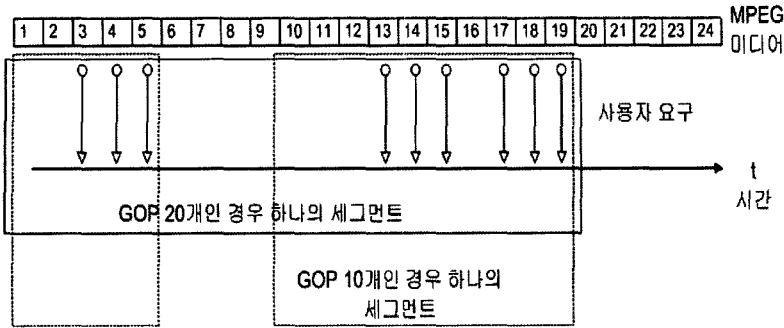


그림 2 참여 GOP 개수에 따른 세그먼트 형태

선호도기반 세그먼트 버퍼 대체(PSBR : Preference-based Segment Buffer Replacement) 알고리즘은 세그먼트를 대체의 기본 단위로 하고 사용자들의 선호도를 각각의 세그먼트별로 유지하여 대체 정책에 사용하는 방법이다. 메모리 공간이 부족하여 기존에 적재된 미디어 데이터들을 대체해야 할 경우 서비스가 되는 스트리밍 미디어 중 선호도가 가장 낮은 미디어를 선택한다. 다음으로 해당 미디어를 서비스하는 세그먼트들 중 가장 낮은 사용자들을 보유하고 있는 세그먼트를 대체 대상으로 선정한다. 이 방식은 세그먼트를 사용하므로 스트리밍 미디어의 순차적 읽기의 특성인 공간적 국부성을 반영하였고, 스트리밍 미디어의 특성상 인기 있는 미디어에 대한 편중되는 사용자들의 선호도를 대체 정책에 반영하므로 시간적 국부성을 이용한 버퍼 대체 알고리즘이다.

3.3.1 PSBR의 자료구조

PSBR에서 사용하는 자료구조는 그림 3과 같다. 해당 자료구조는 쓰레드 안에서 전역 변수 형태로 존재하며 모든 서비스 쓰레드에 의해 참조 될 수 있다. 자료 관리를 위해서 세 가지 종류의 구조체가 이용된다. 첫째는 MPEG 미디어 데이터에 전반적인 내용을 관리하는 미디어 파일 정보 관리 노드이다. 이 노드는 해당 파일을 서비스 받고 있는 전체 클라이언트 수, 해당 파일의 히트율, GOP 정보 노드 관리를 위한 포인터, 세그먼트 리스트를 관리하기 위한 포인터 등을 관리한다.

두 번째는 GOP 정보 관리 노드이다. GOP 정보 관리 노드에는 해당 파일의 위치, 미디어에 대한 고유번호, 할당 받은 버퍼의 시작 인덱스, 사용하고 있는 버퍼 수, GOP 번호, 크기, 위치 등의 정보를 담고 있고 실제 데이터를 디스크로부터 읽어 올 때 이용한다.

세 번째는 세그먼트 정보 관리 노드이다. 세그먼트 정보 관리 노드에는 해당 세그먼트를 참조하는 클라이언트 수, 세그먼트 내 히트율, 사용하고 있는 버퍼 수, 리스트 유지를 위한 포인터 등이 있다.

그림 3에서 보는 바와 같이 모든 노드들은 이중 연결 리스트로 연결되어 사용자 서비스 쓰레드가 생성되거나 삭제될 때 현재 서비스하고 있는 사용자 수를 기반으로 오름차순으로 정렬하게 된다. 먼저 미디어 파일 정보 관리 노드에 대한 정렬이 이루어지고 다음으로 파일 내 세그먼트 정보 관리 노드에 대한 정렬을 수행한다. 서비스의 연결이 이루어져 새로운 클라이언트가 추가 될 때나 연결 해제로 인해 클라이언트가 감소 할 때마다 정렬을 다시 수행한다. 이러한 동작을 통하여 최근의 사용자 정보를 수정하기 때문에 현재의 선호도를 버퍼 대체 정책에 반영할 수 있다. 또한 항상 정렬된 상태를 유지하기 때문에 버퍼 대체 시 대체 대상 노드를 찾고 선정하는데 걸리는 오버헤드를 감소시킬 수 있다.

3.3.2 PSBR의 제어흐름

그림 4는 서비스 쓰레드의 동작과정을 보여주는 프로그램 순서도이다. 서비스 쓰레드는 자신이 서비스하는 미디어 데이터에 대한 정보, 자신이 속한 세그먼트 정보, 현재 서비스하고 있는 GOP 정보 등에 대한 포인터를 가지고 해당 서비스를 하게 된다. 하나의 서비스 쓰레드가 생성될 때 미디어 데이터에 대한 정보와 서비스를 위한 사용자 정보를 실행 인자로 받게 된다. 생성된 쓰레드는 해당 미디어 데이터에 대한 미디어 파일 정보 관리 노드가 있는지 먼저 검사한다. 만약 이 쓰레드가 해당 영화 미디어에 대한 첫 번째 서비스요청이면 미디어 파일 정보 관리 노드를 새로 생성하게 된다. 반면에 이미 해당 영화 미디어가 서비스 중이라면, 자신이 속하게 될 세그먼트를 찾아 자신이 속할 수 있는지를 검사한다. 만약 자신이 이미 존재하는 세그먼트에 세그먼트들의 시간 폭 내에서 속할 수 없다면, 새로운 세그먼트 정보 관리 노드를 생성하게 된다.

3.3.3 세그먼트의 할당 및 반환

PSBR에서는 세그먼트 기준 단위로 하여 메모리를 관리하지만, 실제 세그먼트를 구성하는 메모리는 10KB 단위로 고정된 버퍼들의 집합으로 이루어진다. MPEG 미

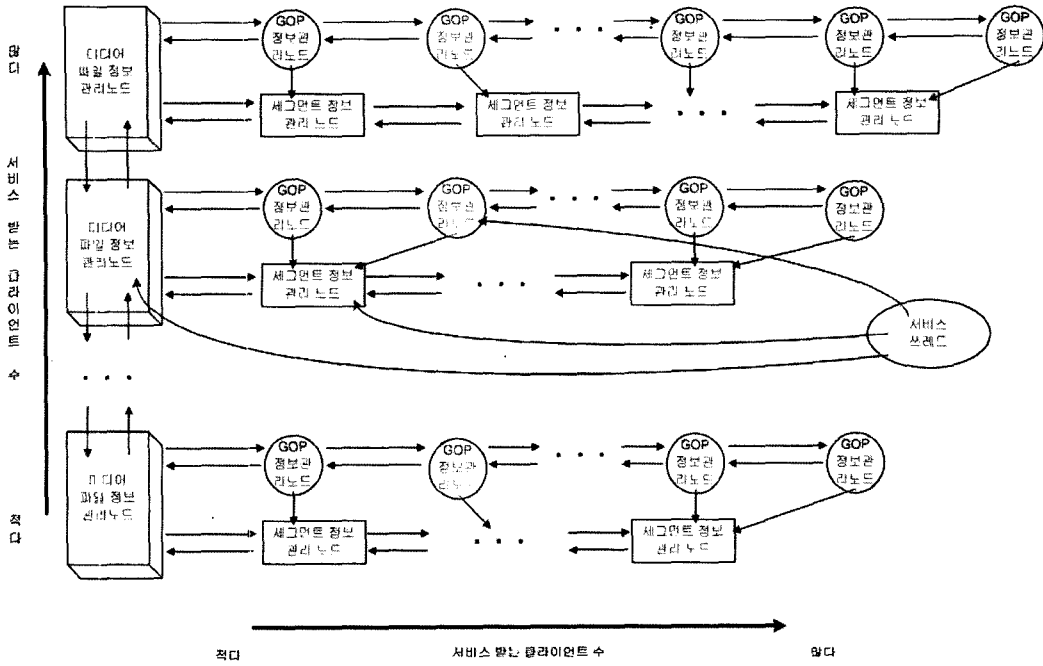


그림 3 PSBR의 전역 자료구조

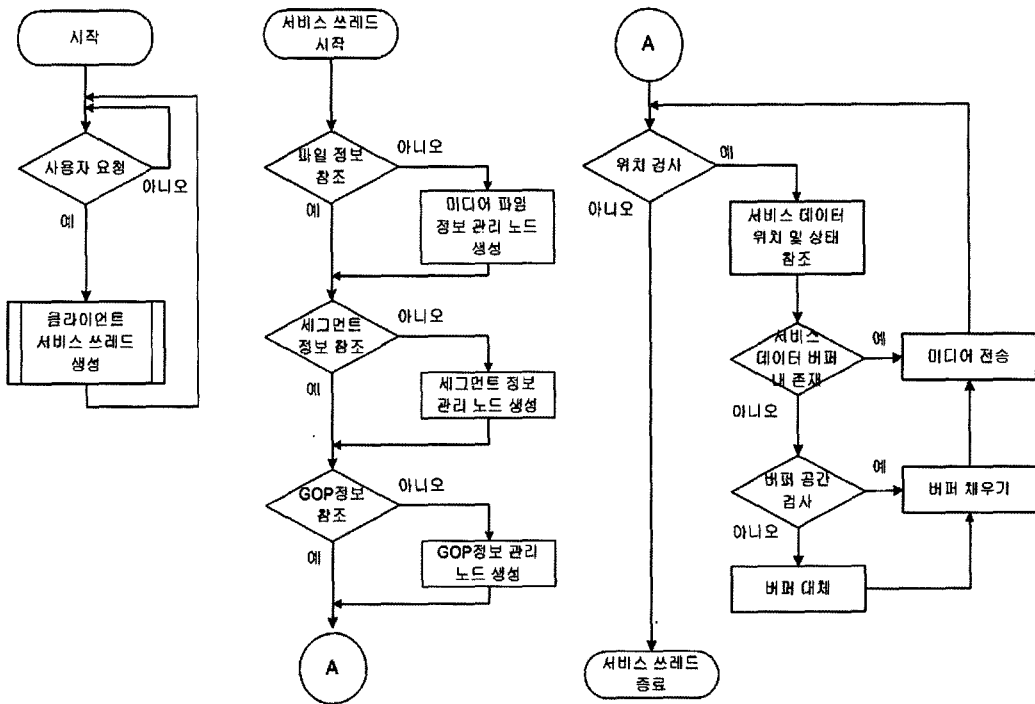


그림 4 PSBR의 제어 흐름도

디어의 GOP 사이즈가 수십 킬로바이트에서 수백 킬로바이트까지 가변적이지만, 프로그램 상에서 네트워크로 전송하는 기본 단위가 10KB이기 때문에 버퍼를 10KB 단위로 나누었다. 버퍼의 기본 단위를 크게 하면 작은 GOP 정보를 저장할 경우 내부 단편화에 의한 메모리 낭비를 초래한다. 반면에 버퍼의 기본 단위가 너무 작은 경우 빈번한 메모리 복사 동작으로 인해 전체적인 성능을 저하시킬 수 있다.

크기가 가변적인 GOP를 관리하기 위해서 버퍼는 다음 부분을 저장하는 버퍼 위치를 알 수 있도록 인덱스를 가지고 있고, 이 인덱스를 통하여 다음 버퍼를 참조

한다. 그림 5는 하나의 서비스 쓰레드를 이루는 주요 구성 요소들을 보여주고 있다. 하나의 서비스 쓰레드는 자신이 속한 미디어 파일 정보 관리 노드, 세그먼트 정보 관리 노드, GOP 정보 관리 노드를 직접 참조 할 수 있는 포인터를 가지고 있어 필요할 때 바로 참조가 가능하다. 하나의 서비스 쓰레드는 현재 자신이 서비스하고 있는 GOP 정보 관리 노드를 통해서 해당 미디어 데이터가 버퍼 내에 있는지 없는지를 알 수 있다. 이 정보를 이용하여 빈 버퍼에 데이터를 채우고 반환하는 작업을 수행 할 수 있다.

그림 6은 서비스 쓰레드에 의해 하나의 세그먼트 내

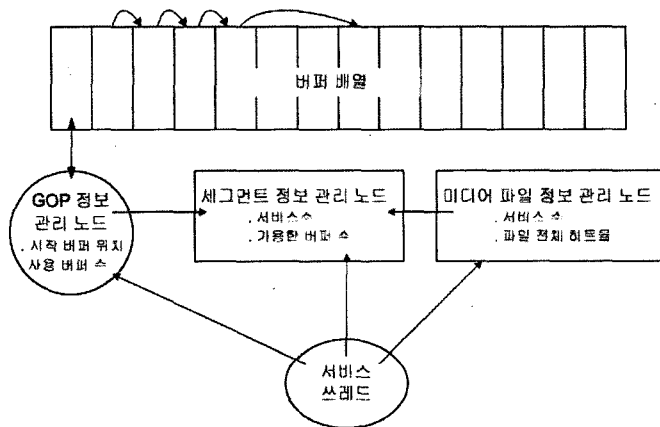


그림 5 서비스 쓰레드의 구성

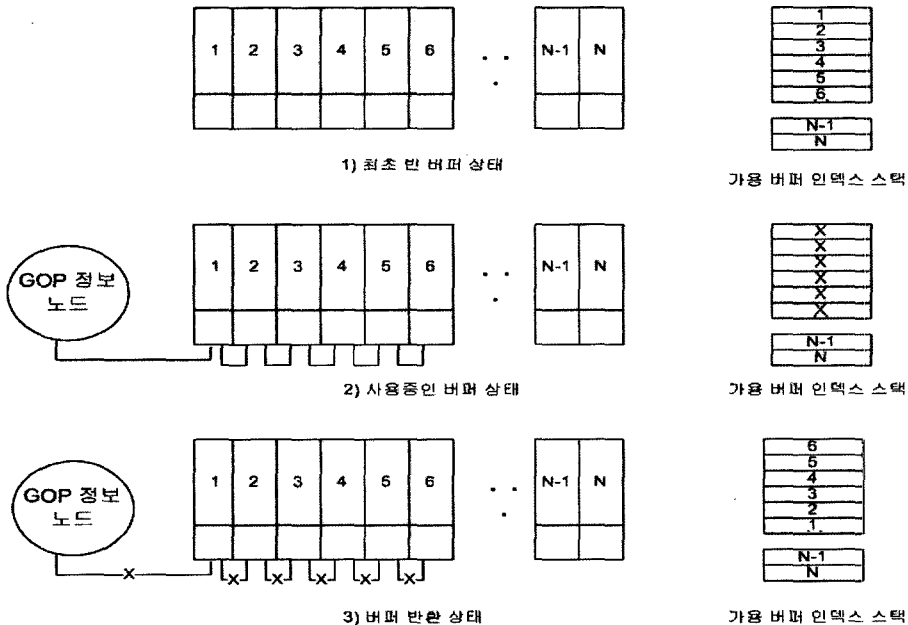


그림 6 세그먼트를 구성하는 버퍼의 할당과 반환

부의 버퍼들에 데이터가 할당 되고 대체 되는 과정을 보여주는 것이다. 전역으로 관리되는 버퍼 공간을 할당 받고 반납하는 과정에서 공간의 사용 유무를 할 수 있도록 가용 버퍼 인덱스 스택을 두었다. 버퍼 공간을 할당 받고자 하는 쓰레드들은 이 스택에서 버퍼 인덱스를 가져와서 해당 공간에 데이터를 채워 넣는다. 이렇게 데이터를 채우게 되면 가용 버퍼 인덱스 스택의 크기는 줄어들게 되고 GOP 정보 관리 노드에는 버퍼의 시작 인덱스를 저장하게 된다. 만약 가용 버퍼 인덱스 스택에 가용한 인덱스가 없으면 PSBR 알고리즘에 기반 하여 선택된 세그먼트에 포함된 버퍼들의 대체가 이루어지게 된다. 하나의 세그먼트가 대체 대상으로 선정되어 공간을 반납해야 할 경우 인덱스를 따라가면서 반납하고자 하는 버퍼들에 대한 인덱스를 가용 버퍼 인덱스 스택에 저장하게 된다. 이렇게 반납된 버퍼 인덱스는 PSBR 정책에 따라 새로운 미디어 데이터들을 위한 버퍼들로서 재사용된다.

4. 실험 환경

4.1 클러스터 VOD 서버

본 논문에서 설계 구현한 클러스터 VOD 서버는 그림 7에서 보는 바와 같이 하나의 HS(Head-end Server) 노드와 다수의 MMS(Media Management Server) 노드들로 구성된다. HS 노드는 사용자의 접속 요청을 받아들이고 일반적인 클러스터링 시스템에서 부하균형을 유지하기 위한 역할을 수행한다. 이를 위해 사용자에 대한 진입 통제 및 미디어 영화 재생 요청을 처리하고, MMS 노드들을 관리, 제어하는 역할을 수행한다. 반면에 MMS는 HS의 제어에 따라 사용자에게 영화 데이터를 전송하여 주는 서버로서 스트리밍 서버 또는 스토리지 서버의 역할을 수행한다. MPEG 미디어의 타입으로 생성된 각각의 영화들은 MMS 노드들에 GOP 단위로 스트라이핑 되어 디스크에 저장되게 된다. 또한 MMS

에는 버퍼 메커니즘들이 탑재되어 각각의 정책에 따라서 디스크에서 읽어 들인 데이터들을 버퍼에 저장, 대체 하는 기능을 수행하게 된다.

4.2 성능 측정기

클러스터 VOD 서버의 성능 측정을 위하여 계측 프로그램(Yardstick Program)을 만들었다[25]. 계측 프로그램은 가상 부하 서버, 가상 클라이언트로 이루어져 있다. 가상 부하 서버는 HS 노드에서 동작하는 성능 측정용 프로그램으로서 가상 클라이언트들에게 영화 미디어에 대한 제어 명령을 전송한다. 가상 클라이언트는 가상 부하 서버로부터 전송된 제어 명령에 따라서 새로운 사용자에게 대한 접속 요청을 HS 노드에게 보내고, HS 노드로부터 승인이 허락된 후 MMS로부터 해당된 미디어의 데이터를 전송 받게 된다. 각각의 가상 클라이언트는 자신이 요청한 미디어의 해당 비트율 수신한 시간을 측정하고, 1초에서 그 시간을 제외한 만큼 usleep() 함수를 이용하여 대기하도록 함으로서 요청한 영화 미디어의 대역폭 요구량에 맞게 비디오 데이터를 소모하여 실제 클라이언트와 동일한 부하를 서버에 준다. 만약 해당된 비트율이 만족하지 못하는 스트림이 발생하면 가상 부하 서버에게 오류를 전송한다. 오류가 발생된 스트림은 현재의 VOD 서버에서 자신의 규격에 맞는 QoS를 하지 못함을 의미한다. 즉, 각각의 영화 미디어 마다 생성될 때 지정된 비트율이 있으며 이 기준을 만족하지 못할 경우 VOD 서비스의 특징상 서비스되는 사용자의 수에서 제외되어야 한다.

4.3 실험환경

구성된 클러스터 VOD 서버, VODCA[4]는 1대의 HS와 6대의 MMS로 구성하였으며, 각 노드의 사양은 표 1과 같이 일반적인 하드웨어와 소프트웨어를 사용하였다. 각 노드는 리눅스를 기반으로 하여 개발되었으며 관리자용 인터페이스를 위하여 Qt 라이브러리를 사용하였고, 그 외 부분들은 C와 C++를 사용하여 개발하였고, gcc 2.96 버전을 이용하여 컴파일 하였다. 데이터베이스는 MySQL을 사용하였다. MPEG 미디어 데이터는 총 10개의 파일을 사용하였으며 목록은 표 2와 같다. 대부분의 영화가 초당 약 143KB 정도의 대역폭을 요구하는 데이터이다.

본 논문의 실험에서는 하나의 세그먼트가 MPEG 미디어내의 연속적인 GOP 15개로 만들어지도록 하여 스

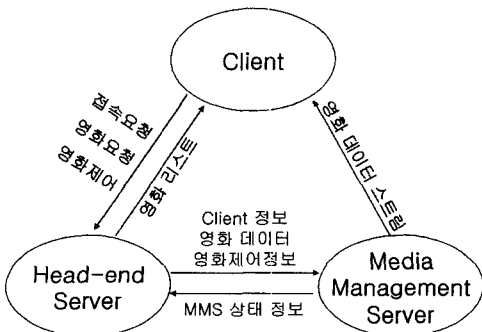


그림 7 클러스터 VOD 서버의 구성

표 1 클러스터 노드의 사양

CPU	Intel(R) Celeron(R) CPU 2.00GHz
메모리	512M SDRAM
운영체제	RedHat 7.3 (Kernel 2.4.18)
네트워크	100Mbps fast-ethernet

표 2 실험에 사용한 MPEG 미디어 데이터

미디어 데이터 고유 번호	제목	해상도	프레임율(fps)	비트율(KB/s)
1	Matrix 3	352×240	29.97	143.8
2	Pay Check	352×240	29.97	143.8
3	Welcome to Jungle	352×240	29.97	143.8
4	Big Fish	352×240	29.97	143.8
5	Lord of the Ring	352×240	29.97	143.8
6	그녀를 믿지 마세요	352×240	29.97	143.8
7	홍반장	352×240	29.97	143.8
8	Spider Man	352×288	25.0	143.8
9	Resident Evil	352×288	25.0	155.8
10	Peter Pan	480×480	29.97	312.5

트리밍 미디어의 특징인 순차적 데이터 읽기에 대한 공간 국부성을 향상하도록 하였다. 또한, 사용자 요청에 대한 빈도는 λ 가 1인 포아송 분포를 이용하여 초당 1개의 요구가 도착하도록 하였으며, MPEG 미디어에 대한 사용자들의 선호 분포는 skew factor가 0.271 zipf 분포를 유지하도록 하였다. skew factor 0.271은 500 번의 사용자 요청 중 130 번이 동일한 미디어에 대한 요청임을 나타낸다.

5. 성능 평가

5.1 버퍼 히트율

제안되는 PSBR 정책과 비교하기 위하여 LRU+Segment 정책을 사용하였다. LRU+Segment 정책은 PSBR 정책과 대등한 환경에서 실험하기 위하여 LRU 정책에 세그먼트 구성을 추가한 것이다. 그림 8은 6개의 MMS 노드들에서 버퍼 대체 정책에 따른 히트율의 평균치를 보여준다. 실험에서는 처음 스트리밍 서비스를 시작한 후 시간이 지남에 따라서 사용자의 수와 사용자가 요구하는 영화 미디어의 종류를 증가시키면서 각각의 MMS 노드에서 발생하는 히트율을 측정한다. 시간이 지남에 따라 서비스 되는 사용자와 미디어의 수가 증가함에 따라 버퍼 메모리에 대한 요구가 급격하게 증가한다. 그 결과 그림 8에서 관찰되는 것처럼 PSBR, LRU+Segment 방식 모두에서 시간이 지남에 따라 히트율이 떨어지고 있음을 알 수 있다.

그림에 나타난 것처럼 초기 시간에는 PSBR 및 LRU+Segment 정책 모두 비슷한 히트율을 보인다. 그러나 시간이 지나서 요구되는 미디어들의 종류가 증가함에 따라 미디어에 대한 선호도 정책을 반영한 PSBR 정책이 LRU+Segment 정책보다 높은 히트율을 나타낼 수 있다. LRU 정책은 과거 참조거리를 기준으로 대체 대상을 선정한다. 하나의 세그먼트 안에 다수의 사용자들이 스트리밍 서비스를 받고 있을 수 있는데 이런

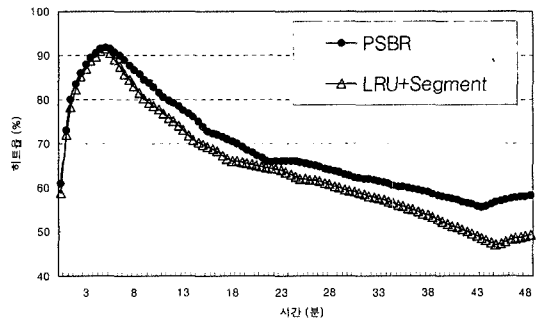


그림 8 버퍼 히트율

경우 그 중 하나에 의해 참조가 이루어지면 최근 참조로 갱신된다. 그러므로 많은 수의 사용자가 있는 세그먼트는 대체 될 확률이 적어지게 된다. 이런 점만 보면 논문에서 제안하는 선호도 기반의 PSBR 대체 정책과 차이가 없다고 생각할 수 있다. 하지만 LRU의 경우는 단지 확률이 적어질 뿐 스트리밍 서비스를 받는 사용자가 많은 세그먼트가 대체에서 배제되는 것을 완전히 보장해 주지는 못한다. 어느 시점에 많은 수의 사용자를 담당하는 세그먼트에 대한 접근이 없을 수 있고, 이때 버퍼의 대체가 필요하다면 해당 세그먼트 내 버퍼는 모두 대체가 될 것이다. 그러나 방금 대체된 버퍼에 대한 사용 요구가 증가하게 될 것이며 그 결과 버퍼 미스율이 증가하게 된다. 이 경우 다시 세그먼트를 생성하고 버퍼를 채우는 비효율적인 작업을 수행해야 한다. 그림 8에서 보는 것처럼 제안된 대체 정책이 LRU에 비해 최고 10%의 높은 히트율을 보여주고 있다.

5.2 성능 확장성

그림 9는 4장에서 제시한 실험 환경에서 MMS 노드 수 및 사용자 수를 증가시키면서 QoS 되는 사용자의 수를 측정하는 결과를 보여주고 있다. 4장에서 제시한 yardstick 계측프로그램을 통하여 해당된 스트리밍 미디어의 QoS가 사용자들에게 공급되는지를 측정하였다. 노

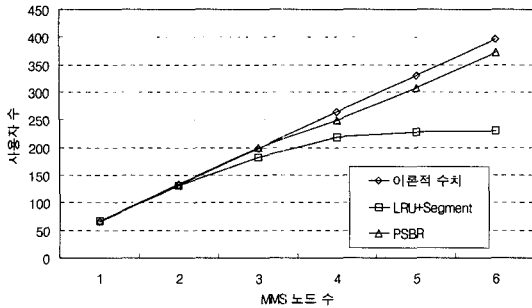


그림 9 버퍼 히트율

드의 증가에 따른 성능 확장성을 시험하기 위하여 구성된 클러스터 VOD 서버에서 점진적으로 MMS 노드 개수를 증가시켜가면서 서비스 되는 모든 스트림들의 지정된 QoS 기준이 만족되는지를 관찰하면서 최대의 서비스가능 사용자수를 측정하였다.

그림 9에서 나타낸 이론적 수치 값을 나타내는 그래프는 실험에 사용된 클러스터 MMS 노드들 개수와 각각의 하드웨어적 규격을 기반으로 계산된 값이다. 실험에 사용된 MMS 노드들은 고속 인터넷 스위치에 연결되어있고 모든 MPEG 영화 미디어는 병렬 처리되어 조개어져 MMS 노드들에 병렬 저장되어있다. 이렇게 분산되어 저장된 MPEG 미디어는 MMS 노드들과 연결된 고속 인터넷 스위치를 통하여 병렬로 동시에 사용자 단말에게 전송된다. 이 경우 네트워크 대역폭이 하드웨어적으로 허용할 수 있는 규격 내에서 가능한 최대 QoS 스트림 수를 이론적 수치로 가정하였다.

그림 9에서 알 수 있듯이 PSBR 정책의 경우 MMS 노드수가 증가할수록 성능 확장성이 이론적 수치 값과 유사하게 선형적으로 증가됨을 관찰할 수 있었다. MMS 노드 6개가 사용될 때 PSBR 정책은 이론적으로 가능한 최대 사용자 수에 비해 약 6% 정도의 성능저하가 나타나고 있다. 이론상의 수치와 실제 서비스 가능 수의 차이는 운영체제나 네트워크 장비 등에서 데이터를 처리하는데 발생하는 오버헤드 때문이라고 생각할 수 있다. 반면에 그림 9에 나타나듯이 LRU+segment 정책의 경우 클러스터 노드 수가 증가함에 따라 성능 확장성은 선형적 증가율이 크게 줄어들고 있음을 알 수 있다. 6개의 클러스터 노드에서 LRU+Segment 정책은 PSBR 정책에 비교 시 약 40% 정도의 떨어진 성능을 나타내고 있다. 이러한 결과의 원인으로는 5.1 장의 그림 8의 버퍼 히트율에서 알 수 있듯이 LRU+Segment 정책은 PSBR 정책보다 저조한 버퍼 히트율을 보이고 있기 때문이다. 즉, LRU+Segment 정책은 스트리밍 미디어의 특성인 사용자 선호도의 편중된 특성을 충분히 반영하지 않으므로 잠시 후에 사용될 버퍼들을 새롭게 요구되

는 사용자들을 위하여 대체 시킴으로 버퍼 미스율이 증가하게 된다. 버퍼 미스는 해당된 데이터를 다시 세그먼트에 채우는 시간을 페널티로 지불해야하므로 MMS 노드의 CPU 는 스트리밍 서비스보다 디스크와 메모리 버퍼의 데이터 이동에 많은 시간을 할애하여야 한다. 그 결과 MMS 노드가 증가할지라도 그림 9에서 관찰되듯이 선형적 성능 확장성을 기대할 수 없었다.

6. 결론

스트리밍 미디어 서비스는 특정 미디어에 대한 편중된 선호도와 순차적 데이터 읽기라는 특징을 가지고 있다. 대규모의 사용자들에게 QoS 가 보장되는 스트리밍 미디어 서비스를 공급하기 위해서는 서버 내부의 자원 관리에 스트리밍 미디어의 특징을 반영하여야한다. 특히, 제한적인 서버내부의 자원들 중 메모리는 버퍼로 사용되며 버퍼에 저장된 미디어 데이터들의 히트율은 서버 시스템 전체 성능에 큰 영향을 미친다.

본 논문에서는 스트리밍 미디어 서비스를 하는 클러스터 서버를 구현하였고, 구현된 서버에서 사용자의 편중된 영화별 선호도와 영화 미디어 데이터의 순차적 읽기 특성을 고려한 선호도기반 세그먼트 버퍼 대체 (PSBR) 알고리즘을 제안하였다. 제안된 PSBR 정책은 사용자들의 편중된 선호도에 기인한 미디어 데이터의 시간적 국부성과 데이터의 순차적 읽기 특성인 공간적 국부성을 활용한다. 본 논문에서는 이러한 두 가지 특성을 반영하여 버퍼 대체 기법을 구현하기위하여 공간적 단위로 세그먼트 개념을 사용하였다. 시간적으로 근접하게 서비스되는 미디어 데이터들을 하나의 세그먼트로 묶어서 관리하며 사용자들의 선호도를 영화별 그리고 세그먼트별 반영하므로 버퍼의 히트율을 높이고자 하였다. 구현된 클러스터 서버에서 기존에 사용되던 LRU 정책과 비교 시 PSBR 정책은 평균 10%의 향상된 버퍼 히트율을 나타내었다. 향상된 버퍼 히트율은 클러스터 서버의 노드개수를 증가함에 따라 선형적인 성능 확장성을 가능하게 하였고, 서버의 선형적 성능 확장성은 대규모의 사용자들에게 QoS할 수 있는 스트림 개수를 보장할 수 있게 하였다.

참고 문헌

- [1] Dinkar Sitaram, Asit Dan, "Multimedia Servers: Applications, Environments, and Design," Morgan Kaufmann Publishers, 2000.
- [2] Prashant J. Shenoy, Pawan Goyal, Harrick M. Vin, "Data Storage and Retrieval for Video-on-Demand Servers," IEEE Fourth International Symposium on Multimedia Software Engineering (MSE'02), pp. 240-245, December 2002.

- [3] Florin Lahan, Irek Defee, Marius Vlad, Aurelian Pop, Prakash Sastry, "Integrated system for multimedia delivery over broadband ip networks," IEEE Transactions on Consumer Electronics, Vol. 48, No.3, pp. 564-565, 2002.
- [4] 서동만, 방철석, 이좌형, 김병길, 정인범, "리눅스 기반의 클러스터 VOD 서버와 내장형에 클라이언트의 구현", 정보과학회논문지 제10권 제6호, pp.435-447, 2004.
- [5] Jung-Min Choi, Seung-Won Lee, Ki-Dong Chung, "A Multicast Delivery Scheme for VCR Operations in a Large VOD System," 8th IEEE International Conference on Parallel and Distributed Systems, June 26-29, pp. 555-561, 2001.
- [6] D.H.C. Du and Y. J. Lee, "Scalable Server and Storage Architectures for Video Streaming," IEEE International Conference on Multimedia Computing and Systems, pp.191-206, June 1999.
- [7] T. Chiueh, M. Vernick, C. Venkatramani, "Performance Evaluation of Stony Brook Video Server," IEEE ICPPW'98, pp.48-66, 1998.
- [8] C. Martin, P. S. Narayan, B. Ozden, R. Rastogi, and A. Silberschatz, "The Fellini Multimedia Storage System," Journal of Digital Libraries, 1998.
- [9] C.C.Aggarwal, J.L.Wolf, and P.S.Yu, "On optimal batching policies for video-on-demand storage servers," Proc. of IEEE ICMCS'96, pp.253-258, 1996.
- [10] Sooyong Kang, Heon Y. Yeom, "Modeling the Caching Effect in Continuous Media Servers," Multimedia Tools and Applications, 23(3), pp 203-224, 2003.
- [11] Sang-Ho Lee, Kyu-Young Whang, Yang-Sae Moon, Wook-Shin Han, "Dynamic Buffer Allocation in Video-on-Demand Systems," IEEE Transactions on PARALLEL AND DISTRIBUTED SYSTEMS, Vol.15, No.6 pp.1535-1551, 2003.
- [12] Nabil J. Sarhan, Chita R. Das, "Caching and Scheduling in NAD-Based Multimedia Servers," IEEE Transactions on PARALLEL AND DISTRIBUTED SYSTEMS, Vol.15, No.10, pp.921-933, 2004.
- [13] J. Gafsi and E. W. Biersack, "Impact of Buffer Sharing in Multiple Disk Video Server Architecture," In Proceedings in the 6th Open Workshop on High Speed Networks, 1997.
- [14] W. Shi, S. Ghandeharizadeh, "Buffer Sharing in Video-On-Demand Server," SIGMETRICS Performance Evaluation Review 25, pp.13-20, 1997.
- [15] 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.
- [16] Donghee Lee, Jongmoo Choi, Jong-Hun Kim, Sam H. Noh, Sang Lyul Min, Yook Cho, Chong Sang Kim, "On the Existence of a Spectrum of Policies that Subsumes the Least Recently Used(LRU) and Least Frequently Used(LFU) Policies," Proc. Of the 1999 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, pp. 134-143, 1999.
- [17] Yannis Smaragdakis, Scott Kaplan, and Paul Wilson, "EELRU: Simple and Effective Adaptive Page Replacement," Proc. OF the 1999 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, pp 122-133, 1999.
- [18] S. Shen, K. A. Hua, and W. Tavanapong. "Dynamic grouping: An efficient buffer management scheme for video-on-demand servers," Technical Report CSTR-97-02, University of Central Florida, Orlando, Florida, 1997.
- [19] 김순철, 조유근, "가변 비트율을 이용하는 주문형 비디오 서버에서의 효율적인 버퍼 관리 기법," 정보과학회 논문지(A) 제25 제2호, pp.177-186, 1998.
- [20] Ethendranath Bommaiah, Katerine Guo, Markus Hofmann and Sanjoy Paul, "Design and Implementation of a Caching System for Streaming Media over the Internet," IEEE Real Time Technology and Applications Symposium, May 2000.
- [21] Asit Dan, Dinkar Sitaram, "Buffer Management Policy for an On-Demand Video Server," IBM Research Repory 19347, Yorktown Heights, NY 1993.
- [22] Asit Dan, Dinkar Sitaram, "A Generalized Onterval Caching Plicy for Mixed Interactive and Long Video Workloads," Proceedings of IS&T/SPIE Multimedia Computing and Networking 1996, Jan. 2006.
- [23] Songqing Chen, Bo Shen, Yong Yan, Sujoy Basu, and Xiaodong Zhang, "SRB: Shared Running Buffers in Proxy to Exploit Memory Locality of Multiple Streaming Media Sessions," IEEE Proceedings of the 24th International Conference on Distributed Computing Systems, 2004.
- [24] 이상호, 문양세, 황규영, 조완섭, "주문형 비디오 시스템에서의 동적 버퍼 할당 기법", 정보과학회논문지 시스템 및 이론 제28권 제9호, pp.442-460, 2001.
- [25] Brian K. Schmidt, Monica S. Lam, J. Duane Northcutt, "The interactive performance of SLIM: a stateless, thin-client architecture," ACM SOSP'99, pp.31-47. 1999.



서동만

2002년 2월 강원대학교 컴퓨터학과 학사
 2004년 2월 강원대학교 컴퓨터정보통신
 공학과 석사. 2004년 3월~현재 강원대
 학교 컴퓨터정보통신공학과 박사 과정. 관
 심분야는 병렬처리, 멀티미디어 시스템,
 운영체제, 센서네트워크



이좌형

2003년 2월 강원대학교 정보통신공학과 학사. 2005년 2월 강원대학교 컴퓨터정보통신공학과 석사. 2005년 9월~ 강원대학교 컴퓨터정보통신공학과 박사 과정. 관심분야는 운영체제, 병렬처리, 파일시스템, 멀티미디어시스템, 센서네트워크



방철석

2002년 8월 강원대학교 정보통신공학과 학사. 2004년 8월 강원대학교 컴퓨터정보통신공학과 석사. 2004년 7월~2006년 2월 (주)씨인씨 연구원. 2006년 3월~현재 리얼메신저 개발팀장. 관심분야는 실시간 시스템, 리눅스, 운영체제



임동선

1986년 숭실대학교 전자계산학과(공학사). 1996년 KAIST 정보및통신공학과(공학석사). 1986년~현재 한국전자통신연구원 임베디드S/W연구단 책임연구원 2003년~현재 강원대학교 컴퓨터정보통신공학과 박사과정. 관심분야는 임베디드 소프트웨어 플랫폼, 실시간 시스템, 소프트웨어 아키텍처



정인범

1985년 고려대학교 전자공학과 졸업(학사). 1985년~1995년 (주)삼성전자 컴퓨터 시스템사업부 선임 연구원. 1992년~1994년 한국과학기술원 정보및통신공학과 졸업(컴퓨터공학 석사). 1995년 2000년 8월 한국과학기술원 전산학과 졸업(박사). 2001년~현재 강원대학교 전기전자정보통신공학부 컴퓨터전공 교수. 관심분야는 운영체제



김운

1993년 8월 고려대학교 전자공학과 학사 1995년 8월 고려대학교 전자공학과 석사 1995년~1999년 LG전자 안양연구소 주임연구원. 2003년 8월 고려대학교 전자공학과 박사. 2004년 3월~현재 강원대학교 교수. 관심분야는 디지털 영상 신호 처리, 멀티미디어 통신, 센서 네트워크