

주문형 비디오 서버에서의 효율적인 버퍼관리 기법

정 흥 기, 박 승 규
아주대학교 정보 및 컴퓨터공학부
E-mail : keeper@madang.ajou.ac.kr

The Efficient Buffer Management Schemes in a VOD Server

Hong-ki Jung, Seung-kyu Park.
Devison of Information and Computer Engineering, Ajou University

요 약

주문형 비디오(Video On Demand) 서버를 구현하는데, 현재까지도 많은 부분에서 한계에 부딪치고 있으며, 이를 해결하기 위한 연구들이 활발히 진행되고 있다. 주문형 비디오 서비스에서의 한계는 대부분이 자원(디스크, 버퍼)의 제한 때문에 발생하고 있으며, 디스크오래이나, 메모리 뱅크와 같은 특수한 하드웨어를 사용하여 해결하고 있다. 또한, 이에 맞추어 여러 가지 소프트웨어적인 방법도 제안되었다. 본 논문에서는 주문형 비디오 서버의 한계 중 디스크대역폭과 버퍼 공간의 한계점에 의한 문제를 해결하기 위하여 기존의 연구분야인 버퍼 공유 및 관리 방법을 제안한다. 제안하는 버퍼 공유 및 관리 방법은 버퍼의 크기 및 디스크의 대역폭 가용량에 따른 적용적 병합, 분할 방식을 사용한다. 제안하는 병합, 분할방식은 피기백킹(piggybacking) 개념을 도입한 방법을 사용한다. 이러한 적용적 버퍼관리의 효율성을 보기 위해 주어진 환경을 설정한 시뮬레이션을 통해 제안방법의 성능을 보였다.

1. 개요

최근, 네트워크와 저장장치, I/O 장치의 발달로 네트워크를 통해 개인이 가정에서 또는 개인 단말기로 멀티미디어 데이터를 사용할 수 있게 되었다. 일반적으로 멀티미디어 데이터는 부피가 크고, 실시간으로 처리가 되어야 하며, 서로 다른 미디어간의 동기화와 한 미디어 안에서의 동기화를 필요로 하는 특징을 가지고 있다. 이와 같은 특징은 멀티미디어 데이터를 네트워크를 통하여 다수의 사용자에게 제공하는 경우에 최적화해야 할 많은 문제점을 발생시킨다. 특히, 주문형 비디오(Video On Demand) 서버를 구현하는데, 현재까지도 많은 부분에서 한계에 부딪치고 있으며, 이를 해결하기 위한 방법으로 그룹핑이나 버퍼 공유기법이 연구되어 왔다. 일반적으로 그룹핑 방법은 사용자의 비디오에 대한 요구를 일정 기간동안 모아서 마치 하나의 사용자 같이 만들어 주는 방법을 말한다.[3] 버퍼 공유기법이라는 것은 어떤 사용자가 요구한 비디오 스트림이 다른 사용자의 요구에 의하여 미리 읽어와서 메모리에 존재할 때, 디스크로부터 스트림을 읽어오지 않고, 메모리내의 스트림을 공유하는 방법을 말한다.[1][2] 버퍼 공유기법에서 논점은 버퍼 공유에 필요한 메모리의 크기를 결정하는 방법과 버퍼를 공유한 그룹간의 병합과 분할 방법이다. 일반적으로 기존의 세션의 병합과 분할 시의 문제점은, 병합을 할 경우 추가의 버퍼공간이 필요로 하게되며, 분할을 할 경우 추가의 디스크 대역폭이 요구된다. 따라서 비디오 스

트림을 관리하는 단위인 세션의 병합과 분할 시에 여분의 자원이 없다면, 병합과 분할을 진행하지 못하게된다.[3] 본 논문에서는 개선된 비디오 스트림의 버퍼 공유방안을 제안한다. 제안하는 버퍼 공유방법은 메모리의 크기를 비디오의 인기도에 기반을 해서 결정하였으며, 자원의 가용량 변화가 불가능한 경우는 약간의 QoS 감소를 감안한 피기백킹 개념을 도입한 세션 병합과 분할을 방법을 적용함으로써 기존의 문제점의 개선을 시도하였다.[4][5]

2. 주문형 비디오 서버의 기본구조

본 논문에서 제안하는 기본적인 주문형 비디오 서버에서 사용자가 요구한 비디오 스트림의 가상적인 흐름은 <그룹 1>에서 나타내고 있다. 본 논문에서는 가상적인 계층을 크게 그룹계층, 세션계층, 그리고 디스크 계층으로 나눈다. 디스크 계층은 물리적인 디스크가 존재하는 곳이며, 각각의 영화는 디스크에 분할(stripe)된 방식으로 저장되어 있다고 가정한다. 세션계층은 물리적으로는 버퍼공간을 의미하며 가상적인 의미에서는 버퍼를 공유하는 그룹을 의미한다. 따라서 하나의 세션은 하나의 디스크 대역폭을 사용하게 되고, 같은 종류의 비디오 데이터를 가지고 있다. 마지막으로 그룹 계층은 그룹이 존재하는 계층으로 각각의 그룹들은 일정한 시간 간격동안 같은 비디오 스트림에 대한 요구가 들어 왔을 때, 마치 하나의 사용자와 같이 합쳐지고, 각각의 그룹 안의 사용자들은 동시에 서비스가 된다. 또한 요구된 스트림은 그룹 안의 사용자들에게 ATM망을 통해 멀티 캐스팅 된다고 가정한다.

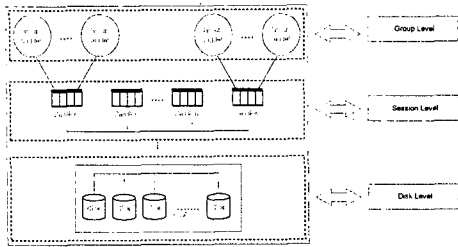


그림 1 스트림의 흐름에 대한 계층구조

3. 버퍼 공유 기법

버퍼 공유 기법은 같은 비디오에 대한 요구들(그룹들) 중, 시간적인 차이가 작은 요구들에 대하여 버퍼를 공유하는 것을 의미한다. 버퍼공유 기법은 디스크의 대역폭의 사용을 줄일 수 있으나, 추가의 버퍼 공간이 필요하다는 단점을 가진다. 본 논문에서는 이와 같은 단점을 완화시킬 수 있는 버퍼 공유기법을 제안한다.

3.1 세션의 크기와 세션에 포함될 조건

세션의 크기를 정하는 것은 일반적으로 세션의 크기를 고정시키는 방법과, 세션의 크기가 가변적으로 변하는 방법이다. 전자는, 서버를 구현하고, 버퍼를 관리할 수 있으나, 시간에 따라 변화하는 사용자의 수에 적응적으로 대처할 수 없으며, 버퍼공간의 낭비도 초래할 수 있다. 따라서 본 논문에서는 후자의 방법을 사용하여 버퍼공간의 낭비를 줄이고 네트워크의 변화에 적응적으로 대처할 수 있는 버퍼 공유 기법을 제안한다. <수식 1>은 i -번째 비디오에 대한 사용자 요구 비율을 나타낸다. λ 는 사용자의 요구가 들어오는 비율이며, 포아송 분포를 따르고 있다고 가정한다.

$$\lambda_i = \lambda f(i) \quad \text{<수식 1>}$$

<수식 2>는 새로운 그룹(요구)이 서비스를 요청하는 시간 간격을 표현하고 있다.

$$t_{inter-arrival_i} = t_{wait_i} + \frac{1}{\lambda_i} \quad \text{<수식 2>}$$

<수식 2>에서 t_{wait_i} 값은 i -번째 비디오에 대한 요구들이 그룹핑하기 위해 기다리는 시간을 의미한다. <수식 3>은 새로운 그룹이 기존의 세션에 포함되어 질 수 있는 시간적인 차이를 나타낸다. C 값은 버퍼의 용량에 관계된 상수이다.

$$t_{join} = \frac{C}{N_{session}} \times \sum_{i=1}^{N_{max}} t_{inter-arrival_i} \quad \text{<수식 3>}$$

3.2 세션의 병합과 분할

3.2.1 세션의 병합

기존의 세션병합의 문제점은 추가의 버퍼공간을 요구한다는 점이다. 따라서 만약 추가의 버퍼공간이 남아있지 않다면, 세션병합은 진행될 수 없다. 본 논문에서는 이러한 문제점을 완화시킬 수 있는 방법을 제안한다. 만약 현재 버퍼 사용률이 낮은 경우에는 기존의 세션 병합 방법을 사용한다. 그러나 만약 두 개의 세션을 병합하려고 하는 시점에서 충분한 버퍼공간의 여유가 없다면, 기존의 방법으로 병합하는 것은 불가능하다. 따라서 본 논문에서는 피기백킹(piggybacking) 개념을 도입한 병합방법을 제안한다. 일반적으로 피기백킹 방법이란, 비디오의 프레임들을 일정 비율로 삭제하거나 삽입하여 비디오 스트림의 재생률을 변화시키는 방법이다. 본 논문에서는 사용

자가 보는 재생률을 빠르게 하는 효과를 수반하는 방법을 사용하여 세션간의 병합을 한다. 이 경우, 두 개의 세션을 피기백킹 방법을 사용하여 병합하는데 소요되는 시간은 <수식 4>와 같이 계산된다.

$$t_{piggy-merge} = \frac{N_{SessionMerge_{i=j}}}{R_{frame-skip}} \quad \text{<수식 4>}$$

본 논문에서 제안하는 방법을 사용하여 세션 병합을 할 경우, $t_{piggy-merge}$ 만큼의 시간이 경과한 후에 병합이 이루어진다. 기존의 병합방법 보다는 시간이 길어지게 되지만, 추가의 버퍼를 요구하지 않는다는 장점이 있다. 반면에 프레임들을 삭제하기 때문에 QoS의 하락이 발생지만, 사람의 인지 불가 범위인 5-8% 사이에서의 범위에서 이루어지고, 지속적인 병합이 아니라 일시적인 병합이므로 기존의 피기백킹 방법에 비하여 심각하게 고려할 문제점은 아니다.

3.2.2 세션의 분할

세션의 분할이란, 동일한 비디오에 대한 요구 중에서 버퍼를 공유하는 그룹을 두 개의 그룹으로 나누는 것을 의미한다. 세션의 분할은 현재 버퍼공간이 부족할 경우 발생한다. 기존의 세션분할 방법은 매우 간단하게 세션을 분할시켜 버퍼공간을 확보할 수는 있지만, 추가의 디스크 대역폭을 요구한다는 단점이 있다. 따라서 만약 추가의 디스크 대역폭이 남아 있지 않다면, 세션 분할은 진행될 수 없다. 본 논문에서는 세션 병합과 마찬가지로, 피기백킹 개념을 이용한 분할 방법을 제안한다. 만약, 현재 디스크 사용률이 낮다면 기존의 분할방식을 사용한다. 따라서 선택된 세션 안에 있는 그룹 중 가장 큰 프레임 차이를 가지고 있는 두 개의 그룹을 찾아내 세션 분할을 적용한다. 그러나 만약 현재 디스크 사용률이 높다면 픽기백킹 개념을 이용한 분할 방법을 사용한다. 제안하는 방법은 현재 서비스 중인 세션들 중에서 내부 그룹사이의 프레임 차이가 가장 큰 세션을 찾아서 뒤에서 오는 그룹의 프레임의 재생률을 빠르게 함으로써, 앞서가는 그룹과 합치는 방법을 사용한다. 두 개의 그룹이 병합이 되면, 기존의 세션 분할과 같은 결과가 발생한다. 그러나 제안하는 방법은 다소 복잡하며 세션 분할을 위해서는 일정한 시간이 필요하다는 단점이 있다. <수식 5>는 세션 분할을 하는데 필요한 시간이

$$t_{piggy-split} = \frac{N_{max-gap_a}}{R_{frame-skip}} \quad \text{<수식 5>}$$

4. 모의실험 및 결과 분석.

4.1 모의실험의 요소

시뮬레이션에서 사용자의 요구가 들어오는 시간 간격은 포아송 분포를 사용했으며, 포아송 분포의 λ 값은 10초이며, 총 버퍼 공간은 1000MB, 디스크 대역폭은 40MB/s로 가정하였다. 시뮬레이션언어로는 C++를 사용했으며 사용자의 요구는 2000번으로 설정하였다

4.2 결과 및 분석

본 시뮬레이션의 주요 비교 방법은 버퍼공간의 사용률과 디스크 대역폭의 사용률에 초점을 맞추었다. 그 이유는 버퍼공간의 사용률이나 디스크 대역폭의 사용률이 높다는 의미는 새로운 요구에 대한 수용이 어렵기 때문이다. 만약 그룹핑과 버퍼공유 방법을 사용하지 않을 경우 버퍼와 디스크 사용률은 비디오의 길이에 의존적으로 변화하게 된다. 따라서 본 논문에서는 언급하지 않겠다.

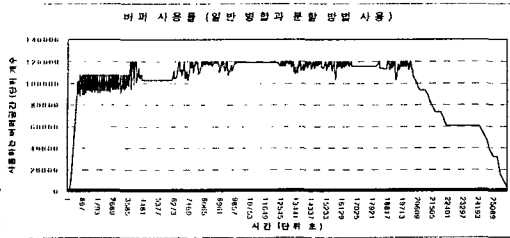


그림 2 버퍼 사용률 (기존의 병합과 분할 방법 사용)

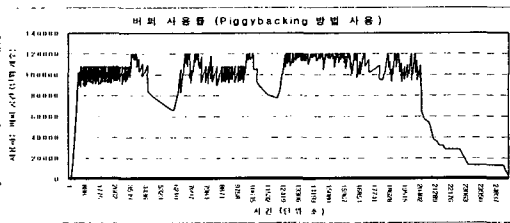


그림 3 버퍼 사용률 (제한한 병합과 분할 방법 사용)

<그림 2>는 일반적인 병합과 분할 방법을 사용했을 경우 버퍼의 사용률을 보여준다. 세션의 병합과 분할로 인해 버퍼의 사용률이 변화가 전체 구간에서 생기는 것을 볼 수 있다. 그러나 이 방법은 어느 정도의 디스크 대역폭이나 버퍼 공간이 남아 있을 경우에만 변화가 심하게 생기며, 그렇지 않을 경우에는 세션 병합과 분할을 진행 할 수 없게 되므로 변화가 점점 없어지며 기존의 서비스가 종료할 때만 새로운 요구를 받을 수 있게 된다. <그림 3>은 기존의 병합, 분할 방법과 제안한 방법을 병행하여 사용했을 경우 버퍼의 사용률을 보여준다. 기존의 버퍼 관리 방법만을 사용한 경우에 반하여 버퍼의 사용률이 높은 경우에도 피기백킹 방법을 이용한 병합과 분할 방법을 이용하여 자원의 더 많은 공유를 할 수 있으므로 버퍼 사용률의 변화가 계속적으로 나타나는 것을 볼 수 있다. 그러나 그래프에서 12000초부터 20000초 정도에서 버퍼 사용률이 높은 상태로 남아 있는 이유는 제안한 방법을 사용한 병합과 분할 방법은 시간을 필요로 하기 때문에, 그 시간 안에 지속적으로 요구가 도착하게 되면 버퍼 사용률은 높은 상태로 남게 된다. 그렇지만 기존의 방법보다는 현재 서버의 버퍼 사용률을 적용 적으로 떨어뜨려 많은 새로운 사용자를 더 서비스 할 수 있다는 점에서 좋은 효과를 보여준다.

<그림 4>와 <그림 5>는 디스크 사용률과 기각된 요구의 수를 보여준다. 이 경우, 디스크 사용률은 앞에서 설명한 버퍼 사용률의 변화와 비슷한 결과를 보여주고 있다. 두 그래프에서 기각된 요구의 수는 기존의 방법을 사용하였을 경우, 297개의 요구가 기각되었으며, 제안한 방법과 기존의 방법을 병행하여 사용하였을 경우, 157개의 요구가 기각되었다. 또한 그림에서 알 수 있듯이 사용자 요구가 기각되는 유형은 버퍼 사용률의 변화보다는 디스크 사용률의 변화에 의존적이며, 이는 실제적인 병목현상은 버퍼보다는 디스크에서 많이 발생한다는 것을 간접적으로 알 수 있다.

5. 결론

본 논문에서는 시간에 따라 변화하는 사용자의 요구와 서버 자원의 사용률에 적응 적으로 대처할 수 있는 세션간의

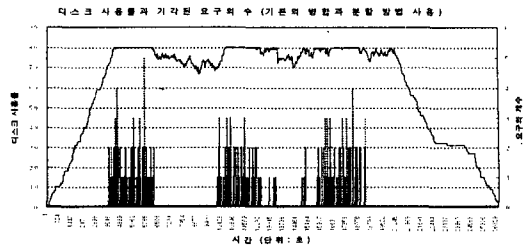


그림 4 디스크 사용률과 취소된 요구의 개수(기존방법사용)

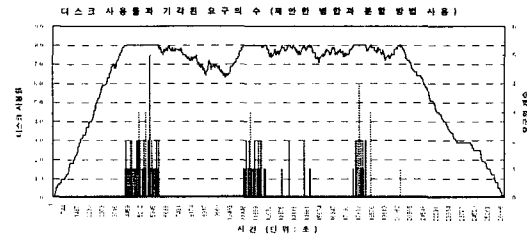


그림 5 디스크 사용률과 취소된 요구의 개수(제안방법사용) 병합과 분할 방법을 제안하였다. 이 방법은 기존에 있던 버퍼 관리 방법의 단점을 보완하여 추가의 자원을 사용하지 않고 병합과 분할을 진행할 수 있으며, 따라서 동시에 보다 많은 사용자의 요구를 서비스 할 수 있는 장점을 가지고 있다.

버퍼 및 대역폭 자원이 동시에 포화상태인 경우 수행되는 피기백킹 방법은 QoS를 일부 떨어뜨린다는 단점을 가지고 있으나, 사람이 인지할 수 없는 5%이내의 가감 상태의 비디오 재현속도 조절로 큰 문제없이 효율적으로 자원을 활용할 수 있다.

실제 환경과 유사한 파라미터로 시뮬레이션을 한 결과, 병합, 분할 방법을 쓰지 않는 경우 곧 포화가 되어 더 이상의 사용자 요구를 수용할 수 없었고, 기존에 제안되었던 병합, 분할 방식을 적용할 경우도 추가의 자원활용이 거의 불가능 하였다.

본 논문에서는 제안한 적응적 자원 공유 방식의 경우 버퍼 및 디스크 사용률에 상당부분의 여부 자원을 확보할 수 있어, 더 많은 사용자 요구를 수용할 수 있었다.

6. 참고문헌

- [1] Asit Dan, Dinkar sitaram, Perwez Shahabuddin "Dynamic batching policies for an on-demand video server", Multimedia systems, pp. 4: 112-121, 1996
- [2] Mohan Kamath, Krithi Ramamritham, Don Towsley, "Continuous media sharing in Multimedia Database Systems", Proceedings of the Fourth International Conference on Database Systems for Advanced Applications (DASFAA'95), April 10-13, 1995
- [3] Wen-Jin Tsai, Suh-Yin Lee, "Dynamic Buffer Management for Near Video-On-Demand systems", Multimedia Tools and Applications 6, pp. 61-83, 1998
- [4] Leana Golubchik, John C.S. Lui, Richard R. Muntz, "Adaptive piggybacking: a novel technique for data sharing in video-on-demand storage servers", Multimedia systems, 4, pp. 140-155, 1996
- [5] Hong-ki Jung, Seung-kyu Park, "Grouping and Buffer Management Methods in a VOD Server", Proceeding of ITCCSCC'99, vol.2, pp. 899-902, 1999.