

고정 선반입과 미디어 Drop을 이용한 VOD 버퍼 스케줄러의 설계 및 분석

문병철^U 박규석
경남대학교 정보통신공학부
{gomdoli, kspark}@kyungnam.ac.kr

Design and Analysis of a VOD Buffer Scheduler Using a Fixed Prefetch and Media Drop

Beong-Cheoul Moon^U Kyoo-Seok Park
Dept. of Computer Science & Engineering, Kyungnam University

요 약

VBR로 압축된 멀티미디어 데이터는 비트 변화율이 매우 큰 편차로 변하기 때문에 자원 예약 관리가 매우 힘들다. 따라서 MPEG데이터의 참조 패턴을 오프라인으로 분석한 메타 테이블을 근거로 과부하 구간에서 미리 선반입하여 시스템의 활용율을 높이는 선반입 기법을 사용하고 있으나, 기존의 선반입 기법은 상영실패는 발생하지 않지만 선반입 임계슬롯이 증가하면서 버퍼 점유량이 증가되어 최소의 적재시간과 적재비용을 유지할 수 없다. 따라서 본 논문에서는 선반입 구간을 고정함으로써 적재비용과 적재시간을 일정한 범위 이하로 유지하면서 시스템 자원의 활용율을 높이는 방법을 제안한다. 그리고 고정 선반입 구간을 사용할 경우 발생하는 상영 실패를 GOP내의 B프레임에 한정하며, Drop모듈을 이용하여 미디어 질 저하를 전체 사용자에게 분산시키는 방법을 제안한다.

1. 서론

멀티미디어 시스템의 일반적인 형태는 멀티미디어 데이터를 저장하고 있는 저장서버와 저장서버에게 원하는 데이터를 요구하는 사용자 시스템 및 이들을 연결하는 통신망으로 구성되는데, 저장 서버가 멀티미디어 시스템의 성능에 큰 영향을 끼친다. 따라서 멀티미디어 데이터를 저장하고 있는 저장 서버에서 효율적인 데이터 관리를 위한 많은 연구들이 진행되고 있다. 그러나 MPEG에 의해 압축된 데이터들은 데이터 내용의 변화로 인해 단위 시간에 재생해야 할 프레임의 크기가 일정하지 않고 이러한 데이터 량의 가변성은 VOD서버에서 자원 예약관리를 매우 어렵게 한다. 이러한 문제를 해결하기 위하여 MPEG으로 압축된 데이터에 대한 참조 패턴을 오프라인으로 분석하여 과부하가 발생할 경우, 미리 선반입하는 선반입 기법을 사용하는데 기존의 선반입 기법에서는 선반입 임계슬롯이 증가하면서 버퍼의 점유량이 증가하여 최소의 적재시간과 적재비용을 유지하지 못한다. 따라서 본 논문에서는 선반입 구간을 고정하여 적재비용과 적재시간을 일정한 범위이하로 유지하면서 시스템 활용율을 높이는 고정 선반입 버퍼 스케줄러를 제안한다. 그리고 제안한 고정 선반입 기법을 사용할 경우 발생하는 상영실패를 GOP내의 B프레임에 한정하며,

Drop 모듈을 이용하여 미디어의 질 저하를 전체 사용자에게 분산시키는 방법을 제안한다.

2. 관련연구

2.1 선반입 정책

선반입 기법은 오프라인으로 분석된 MPEG 데이터 참조 패턴을 이용하여 디스크의 대역폭에 의한 사용자 제한을 서버에 의한 제한으로 전환함으로써 더 많은 사용자를 수용할 수 있다. 즉, 과부하 슬롯에서 디스크 대역폭을 초과한 데이터 요구량을 이전 슬롯에서 여분의 디스크 대역폭을 이용하여 미리 서버의 버퍼로 읽어 들 수 있다. 이때 선반입 되는 데이터 블록들은 사용되기까지 최소한의 시간 동안 적재되어 있도록 선반입 되어야 한다. 데이터 블록의 버퍼 적재 시간을 최소화하면서 과부하 슬롯의 데이터 소비량을 만족하기 위해 선반입 작업을 시작해야 하는 슬롯은 다음과 같이 정의할 수가 있다.

슬롯 t ($\alpha \leq t \leq \omega$)가 영역 $[\alpha, \omega]$ 에서의 과부하 슬롯일 때 식 (1)를 만족하는 최대의 슬롯 ψ ($\leq t$)를 과부하 슬롯 t 의 선반입 시작 슬롯이라 한다.[1]

$$(t - \psi + 1) \times D_{\min} \geq \sum_{i=\psi}^t C(i) \dots\dots\dots (1)$$

2.2 Drop정책

ACME는 캘리포니아 대학에서 제안한 오디오와 비디오 입출력을 제어하는 네트워크 서버로써, 시간 제약을 어기는 경우 전송될 프레임의 일부를 스킵핑(Skipping) 혹은 퍼징(Pausing)하는 기법을 사용하고 있다.[3] HeiTS는 IBM 유럽 네트워킹 센터에서 개발한 멀티미디어 시스템으로 현재의 네트워크 상태를 파악하여 유용한 대역폭으로 트래픽을 적용시키는 네트워크의 응답성에 중점을 두고 있다. 따라서 수신측 시스템의 자원이 과부하 상태로 임계영역에 도달했다면 서비스를 중단하는 것이 아니라 미디어 스트림의 질을 점진적으로 감소시켜 조절함으로써, 사용자에게 의미있는 미디어 스트림을 제공해 줄 수 있다.[4]

3. VOD 버퍼 스케줄링

기존의 선반입 기법은 상영실패는 발생하지 않으나, 선반입 임계슬롯이 증가하면서 버퍼 점유량이 증가하여 최소의 적재시간과 적재비용을 유지할 수 없다. 따라서 본 논문에서는 선반입 구간을 고정함으로써 적재비용과 적재시간을 일정한 범위 이하로 유지하면서 시스템 자원의 활용율을 높이는 방법을 제안한다.

3.1 슬롯 서비스 정책

본 논문에서는 VOD서버가 여러 사용자에게 서비스 수행하기 위하여 필요한 시스템 자원의 할당을 일정한 시간 단위인 슬롯 단위로 자원을 예약, 할당하며 슬롯 서비스 정책은 한 슬롯동안 각 사용자를 위한 자원 예약을 GOP의 배수로 서비스한다.

VOD서버가 현재 n 명의 사용자를 수용하고 각 사용자 C_i ($0 \leq i \leq n-1$)는 서로 다른 영화 M_i 를 검색하고 있을 경우, C_i 를 위해 슬롯 t 동안에 VOD서버가 시스템 버퍼로 읽어 들여야 하는 데이터 량 $C_i(t)$ 는 식 (2)와 같다.

$$C_i(t) = GOP_{m_i} \times k \quad (k=2,4,6\cdots) \quad \text{..... (2)}$$

$C_i(t)$ 는 다수의 GOP로 구성되는데 k 가 클수록 VOD서버는 데이터를 더 오랫동안 시스템 버퍼에 유지해야 하며, k 가 작을수록 승인제어를 위한 알고리즘의 복잡도가 증가하는데 이러한 k 는 시스템 파라미터로 정의할 수 있다. 그리고 VOD 서버가 슬롯 t 동안에 모든 사용자들에 의해 소비되는 데이터 블록의 수 $C(t)$ 는 식 (3)과 같다.

$$C(t) = \sum_{i=0}^{n-1} C_i(t) \quad \text{..... (3)}$$

본 논문에서는 고정된 선반입 임계 구간내에서 요구하는

데이터의 서비스 스케줄링이 완료되어야 하므로 슬롯 서비스는 t 슬롯에서 각 사용자가 요구하는 데이터 량의 총합인 $C(t)$ 에 대한 서비스 스케줄링은 $t-\phi$ 슬롯에서 구성한다. 즉 현재 슬롯이 t 라고 할 때, 현재의 버퍼 상태 $C(t)$ 에 대한 슬롯 서비스 스케줄링은 슬롯 $t-\phi$ 에서 각 사용자가 요구하는 k 개의 GOP 데이터 크기를 추출하여 해당 데이터의 적재, 선반입, 혹은 Drop에 대한 서비스 스케줄링을 구성한다.

3.2 고정 선반입 정책

슬롯 서비스에서 $C(t)$ 를 서비스하면서, 시스템 파라미터인 고정 선반입 임계구간 값 (ϕ)을 더한 $C(t+\phi)$ 에 대한 서비스 스케줄링을 구성하는데, $C(t+\phi)$ 에서 데이터를 적재하지 못할 경우, 식 (4)의 범위 내에서 선반입 가능 슬롯을 선택하게 된다.

$$t+1 \leq \text{선반입 가능 구간} \leq t+(\phi-1) \quad \text{.....(4)}$$

그러나 선반입 구간이 고정됨으로써, 선반입 구간내에 선반입되지 못하는 데이터가 발생할 수 있다. 그러나 초당 24프레임의 영상 데이터가 사용자 앞에 나타났다가 사라지는 상황에서, 연속해서 프레임에 대한 서비스가 되지 않으면 사용자가 감지할 수 있지만, 일정한 간격을 두고 B프레임 1~2개 정도의 상영실패는 사용자가 감지하기 어려울 뿐만 아니라, 설사 감지된다하더라도 허용할 수 실패로 본다.[2]

3.3 Drop 정책

Drop 모듈은 고정 선반입 구간 내에 적재되지 못한 데이터 (θ)가 발생할 경우에 호출되는데, Drop 되는 량을 VOD서버에 서비스 받고 있는 전체 사용자에게 분산시킴으로써, Drop발생이 한 사용자에게 집중하여 발생하는 것과 Drop으로 인한 급격한 미디어 질 저하를 방지한다. 또한 미디어의 질을 보장하기 위하여 I프레임과 P프레임을 Drop 시키지 않으며, GOP내에서 다른 프레임의 복원에 영향을 주지 않는 B프레임만을 Drop시키는데, 특히 B프레임 중 데이터의 크기가 큰 것 ($C(i)_{BF_{max}}$)을 우선적으로 Drop시킨다. B프레임을 Drop시키는 사용자의 수는 θ 의 크기에 따라 정해지는데 식(5)를 만족할 때 까지 사용자의 수를 증가시킨다.

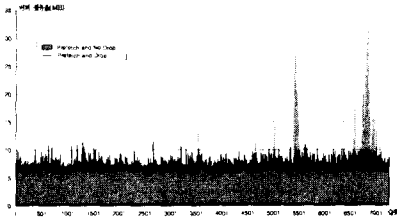
$$\theta \leq \sum_{i=0}^{n-1} C(i)_{BF_{max}} \quad \text{..... (5)}$$

VOD서버에서 Drop모듈이 처음 호출되면 Drop대상자의

User_ID는 1번부터 시작되며, Drop모듈이 한번 이상 호출 되었다면 최근에 Drop되었던 User_ID에 1을 더한 User_ID가 선택되는 라운드 로빈 방식을 사용함으로써, 한 사용자에 대한 집중적인 Drop현상을 전체 사용자에 분산 시킨다.

4. 구현 및 평가

20명의 사용자에게 한 슬롯 당 2 GOP를 서비스하면서 고정 선반입 기법은 선반입 구간을 10슬롯으로 고정시키고, 가변 선반입 기법에서는 선반입 구간을 가변적으로 한 결과 각각의 버퍼 점유량은 그림 1과 같다.

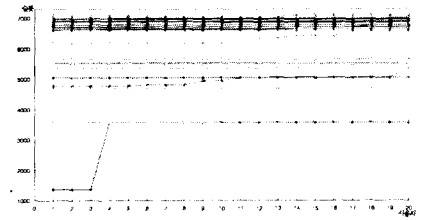


<그림 1> 가변 선반입과 고정 선반입기법의 버퍼 점유량
고정 선반입 기법은 전체 서비스 량의 0.31%를 Drop하면서 버퍼 최대 점유량이 11.42MB까지 증가하는데 비해 가변 선반입 기법은 모든 데이터를 서비스하면서 버퍼 최대 점유량이 33.88MB까지 증가하였다. 이는 가변 선반입 기법은 과부하 발생 구간이 집중적으로 발생하면 선반입 임계구간이 증가되면서 버퍼 점유량이 증가하기 때문이다.

표1. 가변 선반입과 고정 선반입기법의 비교

	가변 선반입 기법	고정 선반입 기법
전체 데이터 요구 량	51.713 GB	51.713 GB
전체 선반입 량	431.8 MB (0.83%)	274.1 MB (0.53%)
전체 Drop 량	-	157.7 MB (0.31%)
선반입 최대 구간	275 Slot	9 Slot
선반입 평균 구간	14.8 Slot	3.1 Slot
버퍼 평균 점유 량	8.23 MB/Slot	7.29 MB/Slot
디스크 대역폭 평균 활용 량	7.13 MB/Slot	7.11 MB/Slot

각 사용자에게 슬롯 당 2 GOP, 선반입 임계 구간 (μ)을 20슬롯으로 서비스한 결과 각 사용자의 데이터 평균 요구량은 50.5 GB, 한 사용자의 평균 데이터 요구량은 2586 MB이며, 총 선반입 량은 338.5 MB(0.65%), 총 Drop량은 93.2 MB(0.18%)이다. 그림2는 각 사용자에게 발생한 Drop 발생간격을 나타낸 것이다. Drop발생간격은 최소 1초에서 최대 36.8분이며, Drop 평균 발생간격은 82.92초임을 알 수 있다.



<그림 2> 사용자별 Drop발생 분포

5. 결론

본 논문에서 제안하는 고정 선반입 기법은 선반입 구간을 고정하고, 선반입 구간의 고정으로 인하여 발생하는 상영실패는 Drop 모듈을 사용하여 상영실패로 인한 미디어 질 저하를 최대한 분산시키는 기법이다. 본 논문에서 제안하는 고정 선반입 기법의 실험 결과, 과부하가 발생하지 않는 상황에서는 가변 선반입 기법과 동일한 성능을 나타내며, 과부하 상황에서는 전체 서비스 량의 0.31%의 상영실패가 발생하지만 가변 선반입 기법에 비해 버퍼 요구량을 $\frac{1}{3}$ 정도로 낮출 수 있는 것으로 나타났다. 또한 각 사용자별 전체 서비스 요구량에 약 0.2%정도의 상영실패가 발생하고 있으나 이는 MPEG복원에 영향을 주지 않는 B프레임에 대한 상영실패이며, 전체 서비스 량의 0.2%정도의 상영실패는 사용자에게 이해되리라 본다. 향후의 연구과제는 선반입 기법에서 MPEG으로 저장된 데이터를 오프라인으로 분석한 대응량의 메타 테이블이 반드시 필요하며, 특히 VOD서버에서 수백편의 영화를 다수의 사용자에게 서비스할 경우 메타 테이블의 운영이 힘들어진다. 따라서 메타 테이블과 고정 선반입 기법을 DB상에서 구현하여 메타 테이블 관리를 수월하게 할 필요가 있다.

참고문헌

- [1] 김순철, "주문형 비디오 서버에서의 효율적인 버퍼 관리 기법" 서울대학교 박사학위 논문, 1998
- [2] 조경운, "라운드를 바탕으로 한 비디오 서버에서의 버퍼 할당 기법" 서울대학교 석사학위 논문, 1997
- [3] R. Govindan, David P. Anderson, "Scheduling and IPC Mechanism For Continupus Media", ACM, selected Area in Computer, pp. 68-80, 1991 pp234-243, May, 1994
- [4] L. C. Wolf, R. G. Herrtwich, "The System Architecture of the Heidelberg Transport System", ACM Operatin System Review, Vol. 28, No. 2, pp51-64, April, 1994